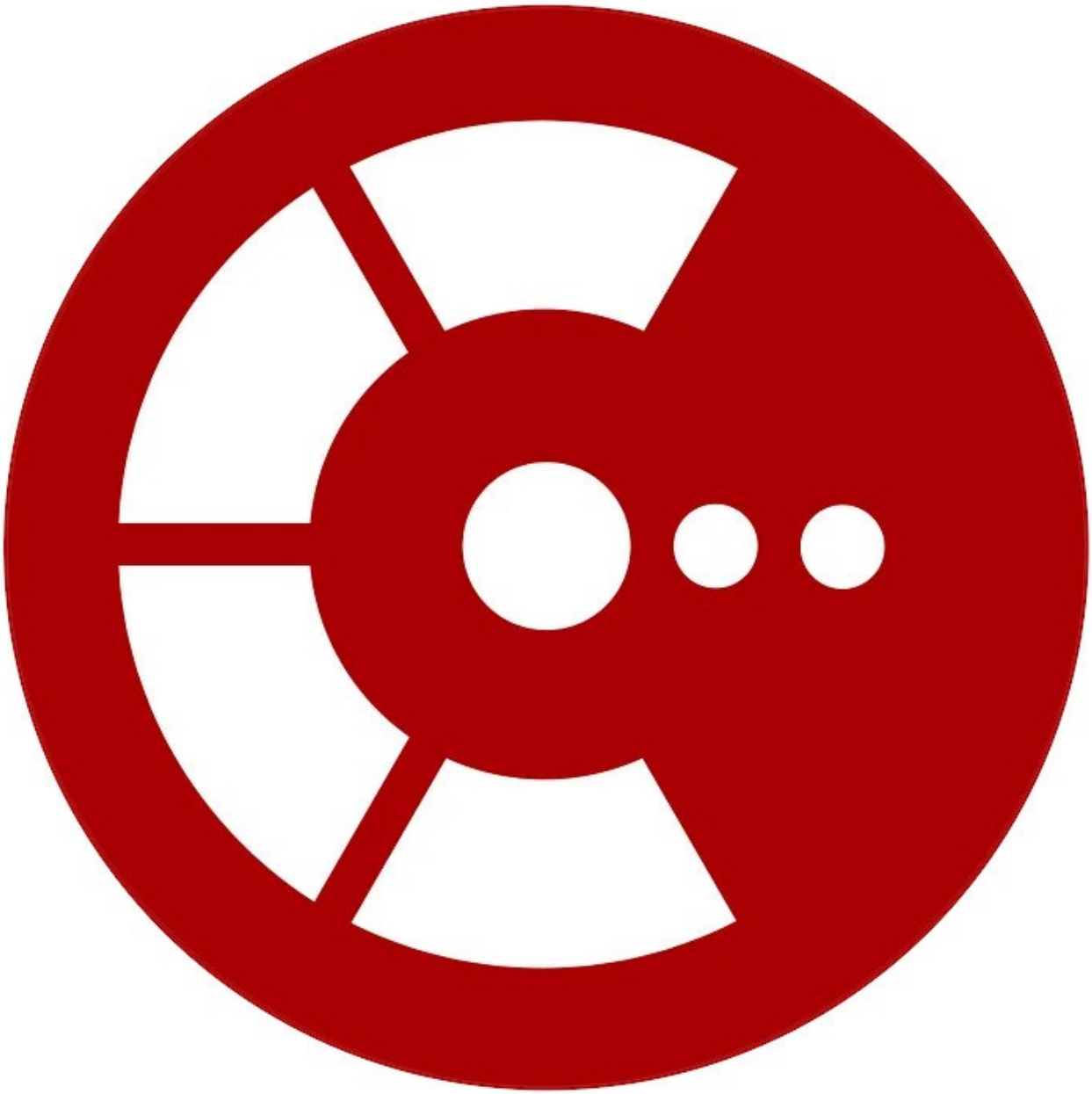


MARMARA KREDİ DÖNGÜLERİ

MARMARA ZİNCİR v.1.1.9

MARMARA CONNECTOR v.0.4.1 Alpha

MCL GELİŞTİRİCİ KILAVUZU

2024

İçindekiler

İçindekiler	2
Marmara Kredi Döngüleri	3
MCL Nedir? Özellikleri nelerdir?	7
MCL Nasıl Çalışır? Nasıl MCL Kazanabilirsiniz?	9
Marmara Connector ile Marmara Blokzincirinin Kolay Kurulumu ve MCL Koin Üretimi	10
Marmara Zinciri Kaynak Kodundan Derlemek	14
Marmara Blokzincirine Başlangıç	19
Marmara Kredi Döngüleri Nasıl Yapılır?	31
Önceden Derlenmiş İkili Dosyalardan Marmara Kurulumu	37
Marmara Blokzincirini Anlamak	40
Marmara Blockchain Staking'in Gücünü Açığa Çıkarın	41
Marmara Chain'de MCL Staking Gücünü Keşfedin	43
Marmara (MCL) Koin Neden Öne Çıkıyor?	45
Marmara Connector	47
MCL Kısa Başlangıç Rehberi	58
Kripto Para Borsalarında İşlemler ve MCL Almak	59
Sıkça Sorulan Sorulara Cevaplar	65
Linkler	69
Kullanıcılara Önemli Uyarı	71
Lisans Detayları	72

Not: İşbu belge Marmara topluluğu gönüllüleri tarafından Marmara Zincir projesinin resmi dökümantasyonundan ve topluluğun Marmara Discord kanalındaki bilgilerden faydalanılarak hazırlanmıştır. Her bir kaynak ilgili konunun altında belirtilmiştir.

Marmara Kredi Döngüleri

MCL (Marmara Credit Loops - Marmara Kredi Döngüleri), gerçek ekonomide çalışmak üzere tasarlanmış ilk ve tek merkeziyetsiz finans sistemidir.

Güvene dayalı/tam güvenli sistemleri ile eşler arası kredi oluşturma ve dolaşım sistemi olan Marmara Zincir'deki Marmara Kredi Döngüleri, ülkemizde ve dünyada yaygın olarak kullanılan vadeli çeklerin dijitalleştirilmiş ilk şeklidir. Marmara Kredi Döngüleri (MCL), Komodo Blockchain teknolojilerinin sağladığı özelliklerin çoğuna sahip olup, Komodo Antara Framework teknolojisi Özel Konsensüs (Custom Consensus - CC) zeki sözleşme yöntemleri ile geliştirilmiştir.

Kredi döngülerinin ilk dijital versiyonu olmasının yanı sıra, yapısı ve blokzincir teknolojisine uygun operasyonları nedeniyle Marmara Zincir (Chain), dünyada blokzinciri bu şekilde uygulayan tek örnektir.

Kredi döngüleri doğrudan blokzincir çekirdeğine uygulanır. Hiçbir şekilde müdahale edilmesi, manipüle edilmesi veya değiştirilmesi mümkün değildir.

Sistemsel özelliklerimiz kısaca aşağıdaki gibidir:

- **En iyi gerçek ekonomide çalışır:** Kredi döngüleriyle alışveriş ederken keşideci ve hamil 3 kat staking ile ödül kazanabilir.
- **Üretilen koinlerin yarısı KİLİTLİ (aktif) olarak çıkar.** Böylece anında staking işlemine girer. Aktif koinleri açabilir veya kredi döngülerinde 3x staking için kilitleyebilirsiniz.
- **Blokların %75'i "Staking" yolu ile üretilmektedir.** Kredi Döngülerindeki koinler 3 kat staking gücü verir.
- **Müdahale Edilemez Kredi Döngüleri:** Kredi döngüleri vadesi dolana kadar 1000 düğümü dolaşabilir.
- **Blok Ödülleri:** Blok zamanı 1 Dakikadır ve blok başına dağıtılan ödül 30 MCL'dir.
- **75% Staking:** Staking ancak kilitli (aktif) koinlerle yapılabilir. Kredi Döngüleriyle 3 kat staking gücünüz vardır.
- **25% Madencilik (PoW) MCL Equihash algoritmasını kullanır.** Komodo ve Zcash'e benzer şekilde madencilik yapabilirsiniz.

Krediye dayalı olarak para yaratmada bankalar yegane kurumlar değildir. Pek çok ülkede insanlar Vadeli Çekler veya senetler aracılığıyla dağıtık şekilde tabanda bankalara benzer şekilde para yaratabilmektedirler.

Kağıt üzerinde analog blokzinciri olarak çalışan bu araçlardan esinlenilen Marmara Kredi Döngüleri (MCL) projesi bunları blokzinciri üzerine taşırken en önemli sorun olan karşılıksızlığı çözmeyi amaçlamaktadır.

İlk sürümde (Versiyon 1.0) karşılıksızlık sorunu blokzinciri kurallarına uygun olarak güvene ihtiyaç duymadan %100 karşılıklılık üzerine çözülmektedir.

İkinci sürüm (Versiyon 2.0) ise karşılıksızlık sorununu güvene dayalı olarak ve topluluk katmanları yardımıyla çözmektedir.

Protokol 1 Milenyum Sürümü:
*Karşılıksızlığa 100% Çözüm
*Güçlendirilmiş Alım Gücü
*Keşideci ve hamilin kredi döngüleriyle alışveriş esnasında 3 katına kadar staking yapabilme gücüne sahip olması

Protokol 2 Güvene Dayalı:
*Sıfır Karşılıksızlık
*Topluluklar aracılığıyla para yaratma
*Eskrovlar aracılığıyla kişilerin kendi topluluklarında kredi oluşturabilmesi

Marmara v.1.0

100% Tam güvenli Blokzinciri Kredi Döngü Sistemi

Protokol 1'de, tüm krediler, kredi veren kişiler tarafından tamamen teminatlandırılır. Bu nedenle, temerrüt riski yoktur. Bunu sağlaması nedeniyle, keşideciler kredi döngülerinde kredinin vadesi tamamlanana dek geçen sürede 3 katına kadar stake edebilme hakkına sahip olurlar.

Alım Gücünün Güçlendirilmesi
Karşılıksızlığa tam çözüm sunan Versiyon 1'de sorunun çözümü kredi döngüsünün her adımında %100 teminat ile gerçekleştirilmektedir. Krediyi ilk ortaya çıkaran keşideci ve sonrasında krediyi döndüren diğer cirantalar %100 teminatı sağlamanın karşılığında blokzincirinde 3 katı "staking" ile ödüllendirilmektedirler. Kredi döngüsüne yeni katılan her ciranta ile döngünün her anında %100 karşılık payı keşideci ve cirantalar arasında risk azaltacak şekilde paylaşılmaktadır.

MCL, kilitli koinler üzerinden kredi dolaşımdayken ilgili kilitli koin sahiplerine stake etme imkanı sağlayan dünyadaki tek stake sistemidir.

Staking Ödüllerinin Tahmini
Aşağıdaki formül kullanılarak staking ödülleri tahmin edilebilir. Formülün yalnızca, hem etkinleştirilen hem de kilitlenen döngü paralarının sayısı ve miktarı, bunların eskimesi, internet bağlantısı, sabit disk hızı, çekirdek sayısı vb. gibi parametrelere göre değişebilecek tahmini sonuçları verdiğini lütfen unutmayın. 2 çekirdek, 4 GB RAM, SSD Harddisk ve makul bir internet bağlantısı ile minimum donanım gereksinimi düşüktür. Hatta bir miniPC ile kolaylıkla staking yapabilirsiniz.

$$\left(\frac{\frac{Activated}{TotalActivated} + \frac{Loops}{Locked\ in\ loops} x\,3}{4} \right) x\,30\,x\,60\,x\,24\,x\,0.75$$

Aktif (Activated): Aktif miktarı (cüzdanınızdaki)

Döngüler (Loops): Döngülerde kilitlenen (cüzdanınızdaki)

*ToplamAktif (TotalActivated): Toplam aktif miktarı

**Döngülerde Kilitlenen Toplam (Locked in loops): Döngülerde Kilitlenen Toplam miktar

30: Dakikada eklenen MCL

60: Saatte yeni eklenen MCL blok sayısı

24: 24 saat

0.75: Toplam staking oranı (%75)

*,** : Güncel “ToplamAktif” ile “Döngülerde Kilitlenen Toplam” miktarları görmek için buraya tıklayabilirsiniz

Marmara v.2.0

Güvene dayalı Kredi Döngüleri

Eskrov sistemi ve eskrov hizmetlerine dayalıdır.

Karşılıksızlığa Karşı Güvene Dayalı Blokzincir Çözümü

Güvene dayalı blokzincir çözümü (Versiyon 2) henüz mevcut blokzincir üzerinde tam olarak uygulanmamıştır. Ancak gereksinim tanımlamaları bitirilmiştir. Güvene dayalı sistem kâğıt bazlı vadeli çeklerdeki mevcut güven temelli analog blokzinciri ile benzer şekilde çalışmakta, ancak geri ödenmeme (temerrüt) ya da karşılıksızlık sorununa karşı birkaç önleyici programla (3 katman önerisi) özgüvence sağlanmaktadır. Güven temelli blokzinciri sisteminde, keşideci bankacılık sistemine benzer bir şekilde kredi yaratır. Buradaki fark, blokzincirinde yaratılan kredinin, genel olarak kredinin banka ile borç alan arasında sözleşme olarak kaldığı banka kredisinin aksine bir kredi döngüsünde dolaştırılmasıdır. Keşideciden başlayarak hamilden önceki düğüme kadar olan tüm düğümler (cirantalar), borç sahibinin yani hamilin borcunun ödenmesinden zincirleme şekilde (müteselsilen) ve kollektif olarak sorumludurlar.

Temerrüde yani karşılıksızlığa karşı çeşitli koruma katmanları vardır. Her şeyden önce, her bir düğüm büyük olasılıkla aktif koinlere, yani temerrüt durumunda teminat olarak kullanılabilecek kilitlenmiş paralara sahiptir. Herhangi bir ciranta veya keşideci düğümün arkasında avalistler olabilir.

Avalistler de kredinin geri ödemesine karşı teminatlandırılan aktif yani kilitli paralarla krediyi geri öderler. Geri ödemenin yapılmaması durumunda kullanılacak fiziksel teminatlar da olabilir.

Karşılıksızlığa karşı nihai çare olarak bir blokzinciri fonu da kullanılacaktır.

Eskrov Sistemi

Güvene dayalı modda, kredi döngülerinin başlatılması veya sonlandırılması dağıtık noterler olarak çalışan güvenilir emanet hizmeti sunucuları tarafından gerçekleştirilir. Bu aracılık hizmeti verenler “Eskrov” olarak isimlendirilirler. Kredi döngüsündeki tüm düğümler, kredi döngü sistemine katılmak için verilerini bir emanet hizmeti (eskrov) tarafından doğrulamalıdırlar.

Eskrov Hizmetleri

Salt madenciler ya da kilitleyerek ödül kazananlar (staking), verilerini Marmara blokzincirinde ilişkilendirmeden de rahatlıkla kullanabilirler. Veri girişi en kritik süreçtir. Oracles’a herkes veri ekleyemez. Veri girişi emanet yoluyla olmalıdır. Emanetler web tabanlı emanet hizmetleri, kripto borsaları, noterler, postaneler, hukuk büroları, kredi birlikleri ve hatta bankalar tarafından olabilir.

Emanetler (escrow) ve avalistler blokzinciri içinde ilave güven katmanı sağlarlar. Halen vadeli çeklerin analog versiyonunda, çekler bankalar tarafından başlatılmakta ve bankalar herkese çek vermemektedirler. Böylece hem veri hem de manuel çözüm süreci sağlarlar.

Sistemde her isteyen eskrov olamaz. Eğer eskrov olmak istiyorsanız lütfen bizimle iletişime geçiniz.

Versiyon 2'de kimlik hırsızlığı olmadığından emin olmak için sadece emanetler yoluyla blokzinciri için Oracles verisi sağlanmalıdır. Bu, emanetler tarafından Müşterinizi Tanıyın (Know Your Customer) prosedürü ile gerçekleştirilecektir. Güvene dayalı sistemin, blokzincirinde Oracles fonksiyonlarını kullanması gerekir.
****
Kaynak: https://marmara.io/marmaracreditloops#marmara_version

MCL Nedir? Özellikleri Nelerdir?

MCL, vadeli çekler ve senetlerin çalışmasından esinlenen yerli bir blokzinciri, yerli kripto para projesidir. Bilmeyenler için söyleyelim. Vadeli çekler bankaların çekleri kullanılarak yapılmasına rağmen bankacılık sistemine alternatif faizsiz bir kredi sistemidir. Esnaf ve işadamları mal ve hizmet satarlarken veya satın alırken vadeli çek araçlarıyla kredi çıkararak ekonomide alternatif bir kredi sistemi çalıştırmaktadırlar.

KARŞILIKSIZLIK VE DİĞER SORUNLAR

Vadeli çekler ve senetler güzel araçlar olmasına rağmen en büyük sorunları karşılıksızlık sorunudur. Güvene dayalı olan bu sistem özellikle ekonomik kriz zamanlarında domino etkisiyle birbiri üzerine çökmektedir. MCL, bu kredilerdeki karşılıksızlık sorununu çözmek için geliştirilen bir projedir.

MCL TEMEL ÖZELLİKLER

MCL'nin açılımı, İngilizce olarak Marmara Credit Loops yani Marmara Kredi Döngüleri'dir.

MCL veya Marmara Koin gerçek bir kullanıma dayalı ilk yerli Türk koinidir. Güzel yanı ise evrensel olarak dünyanın her yerinde ve uluslararası ithalat ihracat vs ödemelerinde rahatlıkla kullanılabilir.

Kripto paralar bildiğiniz gibi henüz gerçek ekonomide mal ve hizmet alışverişlerinde yaygın şekilde kullanılmamaktadır. MCL ise gerçek ekonomide kullanılmak üzere geliştirilmiştir. Kullandığınızda gücünü göreceksiniz.

MCL, Vadeli Çekler ve senetlerin çalışma şeklinden esinlenen bir proje olup çıkışı 2013 yılına kadar gitmektedir. Ufuk Hattı programı sırasında anlatılan Kasaba Hikayesi' ne kadar gitmektedir.

Açık Kaynak ve Uluslararası bir projedir. Blokzinciri Dünyasında yazılım geliştirmek için Blokzinciri Linux'u sayılan Komodo teknolojileri kullanılarak geliştirilmiştir.

2013 yılından başlayarak Marmara Üniversitesi ve başka üniversitelerden akademisyen ve iş dünyasından gelenlerle üniversitede 4 adet çalıştay yapılarak ihtiyaçlar tanımlandıktan sonra yazılım geliştirme yapılmıştır.

Marmara Üniversitesi Mühendislik Fakültesi Açık İnovasyon Laboratuvarı ve Komodo Platform ile birlikte geliştirilen projenin test zinciri çalışmaları 2019'da gerçekleştirilmiş ve zincir Marmara Blokzinciri adıyla 17 Ocak 2020 tarihinde hayatına başlamıştır.

MCL, Türkiye'nin İş Kanıtı ve Hisse Kanıtına dayalı olarak madenciliği yapılan ilk Açık Kaynak yerli kripto para projesi olup gerçek ekonomi üzerinde çalışmak üzere geliştirilmiştir.

MCL veya diğer adıyla Marmara Koin herkes tarafından alıcı ve satıcıların alışveriş içerisinde ödül kazanmaları üzerine kurulmuştur.

MCL basit bir MiniPC üzerinde üretilebilir: Diğer madenciliği yapılan Bitcoin vs sistemlerinde olduğu gibi Enerji tüketimi ve pahalı bilgisayar ihtiyacı yoktur. 2. el bir PC, günlük çalışmalarınızda kullandığınız bilgisayar veya özellikle bu iş için satın aldığınız mini bir PC ile bile üretilebilir.

Ürettiğiniz MCL'leri Merkezsiz Borsa AtomicDEX üzerinde Komodo, Bitcoin ve diğer başka

paralarla dönüştürebilir, piyasada doğrudan mal ve hizmet alımında, alım satımı yapan borsalar veya başka yerlerde Liraya çevirebilirsiniz.

****

Kaynak: https://marmara.io/tr/blog/marmara-kredi-donguleri-1/mcl-nedir-ozellikleri-nelerdir-10

3 Kat Staking Gücü: Kredi döngülerinde kilitlemelerde 3 katına kadar staking gücü

Dakika Başı Blok Ödülü: Blok zamanı 1 dakikadır ve blok başına dağıtılan ödül 30 MCL'dir.

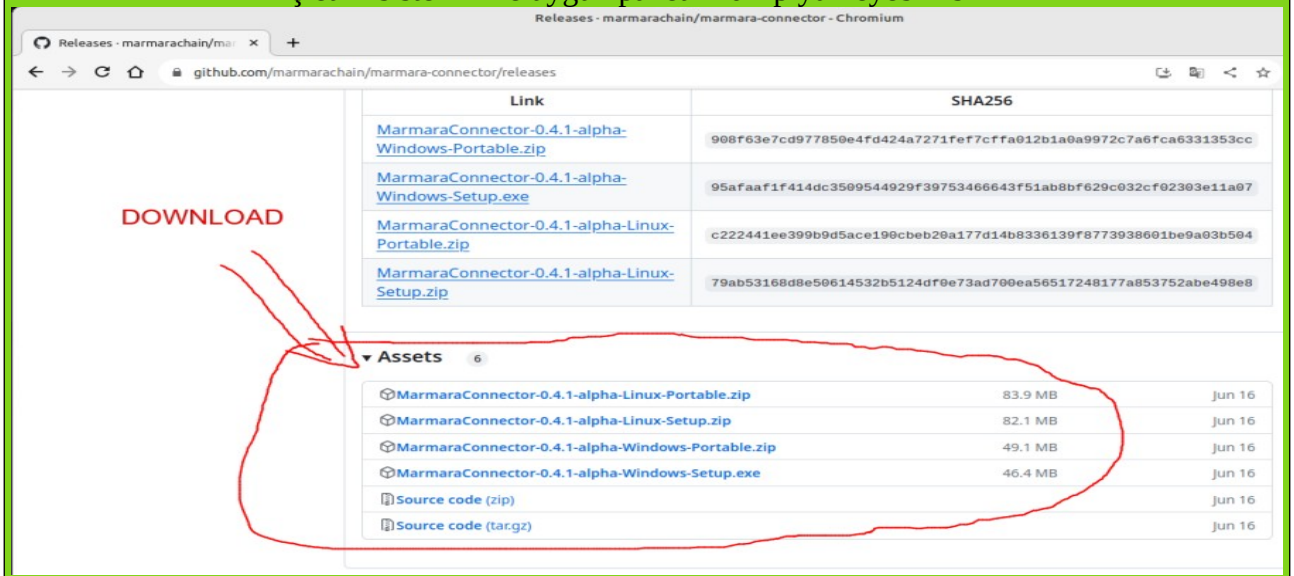
Ön-çıkarım: 2,000,000 MCL

Yıllık Arz : ~ 15,778,800 MCL

ÖNEMLİ BİLGİ

Marmara Zincir kurulumunun en basit yolu, Marmara Connector'ün en son sürümünü kullanarak kurulum yapmaktır. Marmara Connektor, sisteminizde Marmara Zincir'in kurulu olup olmadığını kontrol edecek, kurulu değil ise onay vermeniz halinde kurulum işlemini gerçekleştirecektir. Eğer işletim sisteminize Marmara Connector'ü kurmadan önce Marmara Zincir'i kurduysanız bağlanacak ve grafiksel arayüz üzerinden MCL işlemlerinizi kolaylıkla gerçekleştirmenizi sağlayacaktır. Kurulum sürecinde herhangi bir sorunla karşılaşırsanız discord veya telegram gruplarından yardım ve destek alabilirsiniz.

İşletim sisteminize uygun paketi indirip yükleyebilirsiniz



https://github.com/marmarachain/marmara-connector/releases

MCL Nasıl Çalışır? Nasıl MCL Kazanabilirsiniz?

MCL NASIL ÇALIŞIR?

MCL'nin çalışma şekli çeşitli makalelerde açıklanmıştır. Ama bunu herkesin anlayacağı şekilde açıklayalım.

MCL çalışma mantığı çok basittir. MCL kazanmanın yani MCL üretmenin en kısa ve en doğal yolu alışveriş sırasında kredi döngülerinde kullanmaktır.

ALIŞVERİŞ YAPARKEN KAZANIN

Diyelim ki alıcı veya satıcı olarak başka bir para birimi kullanarak alım yapacaksınız. Peşin ödeme yaptığınızda işlem orada kapanır.

Kredi döngüleriyle MCL ile alışveriş yaparsanız hem alıcı hem de satıcı olarak kazanırsınız. Örneğin reçine ile hobi ürünleri yapıp satıyorsunuz. Alıcıya MCL kredi döngüsü ile satabileceğinizi söylersiniz. Müşteri peşin öderse hiçbir şey kazanamaz. Teminat olarak %100 ödemeyi Marmara blokzincirine kilitlersiniz. Kredi döngüsü vade tarihine kadar 3 kat staking gücü ile MCL ödülleri kazanabilirsiniz. Diyelim ki kredinin vadesinin bitmesini beklemek istemiyorsunuz. Yani nakite ihtiyacınız var. Reçine için malzeme alacaksınız. Kredi döngüsünü tedarikçiye aktarabilirsiniz (buna ileri tarihli çeklerde cirolama denir). Artık müşterinizle birlikte reçine malzemesini satan kişi 3 kat stake ödülü kazanmaya devam edebilir. Ayrıca başka bir müşteri ile yeni döngüler oluşturabilirsiniz.

TÜKETİCİLERİN SATIN ALMA GÜCÜNÜN GÜÇLENDİRİLMESİ

MCL kredi döngüleri (MCL), tüketiciyi yani müşteriyi destekleyen bir sistemdir ve MCL, müşterinin satın alma gücünü pekiştirir. Müşteri peşin öderse, hiçbir şey kazanmaz. Ancak bunu kredi döngüsüyle yaptığında 3 kat staking ödülüne katılma şansı kazanır. Satıcı için de faydalı bir sistem; çünkü artık satın alma gücü yüksek müşterilere sahip bir üreticisiniz. Mal veya hizmetleri daha kolay satabilirsiniz.

MCL kredi döngüleri yöntemini kullanarak taksitli alışveriş yapmaya karar verdiğinizde, alıcı (keşideci) parayı nakit ödeme yapmak yerine, alıcı ve satıcı arasında mutabık kalınan vade ve taksitlerle Marmara Blokzincirine kilitler. Vade sonuna kadar her bir kredi döngüsü için hem alıcı hem de satıcı 3x staking gücü ile Marmara Blokzincirinde blok ödülleri bulma şansı yakalar. Bitcoin madenciliğini duymuşsunuzdur. Bir nevi noterlik hizmeti yaparak Bitcoin olarak ödül kazanıyorlar. Bitcoin ödülleri kazanmak için enerjiye ve pahalı bilgi işlem gücüne ihtiyaçları var. Marmara üzerinde MCL ödülü kazanmak için alıcı ve satıcılar alışveriş sırasında kredi döngüleri yaparak noterlik yapmaları yani kayıt tutmaları yeterlidir.

Bu nedenle MCL, Marmara Kredi Döngüleri, alışveriş sırasında kullanılmak üzere reel ekonomide çalışacak şekilde tasarlanmış Dünyada ilk ve tek DeFi olarak evrensel bir sistemdir. Alışveriş sırasında MCL Kredi döngülerini kullanarak hem alıcı hem de satıcı MCL Coin ödülleri kazanır.

****

Kaynak: https://marmara.io/tr/blog/marmara-kredi-donguleri-1/mcl-nasl-calsr-nasl-mcl-kazanabilirsiniz-9?anim=

Marmara Connector ile Marmara Blokzincirinin Kolay Kurulumu ve MCL Koin Üretimi

Marmara Connector, herkesin kolayca Marmara Blokzincirini kurması ve MCL koin üretmeye başlaması için geliştirilmiş bir uygulamadır.

MARMARA BLOKZİNCİRİ VE MCL NEDİR?

Marmara Blokzinciri Vadeli Çekler ve Senetler gibi tabanda alışveriş içerisinde üretilen kredi araçlarından esinlenmiş bir projedir. Marmara Blokzinciri üzerinde her tür krediyi karşılıksızlık, sahtecilik, bölünemezlik, takip edilemezlik gibi sorunlar olmadan güvenle çıkarabilir (keşide edebilir) ve dolaştırabilirsiniz (cirolayabilirsiniz). MCL gerçek ekonomide kullanmak üzere geliştirilmiş bir sistem olduğundan alışveriş sırasında MCL üretebilirsiniz.

Marmara Blokzinciri özgür bir blokzinciri olup MCL isimli kendi koinine sahiptir. Marmara Blokzinciri üzerindeki MCL bir token değil merkeziyetsiz bir koindir. Marmara Enerji ve işlemci gerektirmeyen %75'lik Hisse Kanıtı algoritması (PoS) nedeniyle gereksinimler mütevazidir. Bu yüzden Marmara Blokzinciri günümüzdeki sıradan bir laptop, PC ve hatta miniPC üzerine kurulabilir ve üzerinde MCL koin üretilebilir.

MCL koin olduğundan bir çeşit noterlik sistemi olarak çalışır. Marmara blokzinciri üzerinde noterlik yaparak ödülleri kazanabilirsiniz. Yaklaşık her dakikada bir blok çıkar ve blok ödülü olarak çıkan 30 MCL noterlik yapma karşılığı dağıtılır. Kredi döngüleri yaparak veya aktifte kilitleyerek bilgisayarınızı açık tutarak kayıtları tutmanız sayesinde blok ödülü kazanabilirsiniz. Ödülü alabilmek için kredi döngülerinde MCL kilitlerseniz 3x staking ile maksimum ödül bulma şansı elde edersiniz. Kredi Döngüleri alışveriş içerisinde cirolanarak (başka bir satıcıya aktararak) alışveriş sırasında kullanılabilir.

MARMARA BLOKZİNCİRİ KURMAK İÇİN GEREKSİNİMLER

Marmara Blokzinciri kurmak için gereksinimler çok düşüktür. Aşağıdaki özelliklerin olması yeterlidir:

- 64 Bit İşletim Sistemi
- 4 GB RAM
- En az 2 Çekirdekli Donanım (4 çekirdek tavsiye edilir)
- SSD 100 GB Harddisk

MARMARA CONNECTOR NEDİR?

Marmara Connector, herkesin kolayca Marmara Blokzincirini kurması ve MCL üretmeye başlaması için geliştirilmiş bir uygulamadır. MCL ödülü kazanmak için blokzinciri kurmak gerektiğinden blokzincirinin herkesçe kolay şekilde kurulması gerekir. İşte Marmara Connector bu amaçla geliştirilmiştir.

Marmara Connector kurulumu ve kullanımı çok basittir. Windows ve Linux İşletim Sistemi üzerinde yerel olarak makineniz üzerinde veya uzaktaki bir sunucuya kurularak çalıştırılabilir.

MARMARA CONNECTOR KURULUMU

Marmara Connector Kurulumu çok basittir.

Marmara Chain Github Sayfasından en son Marmara Connector uygulamasını indirin ve kurun.

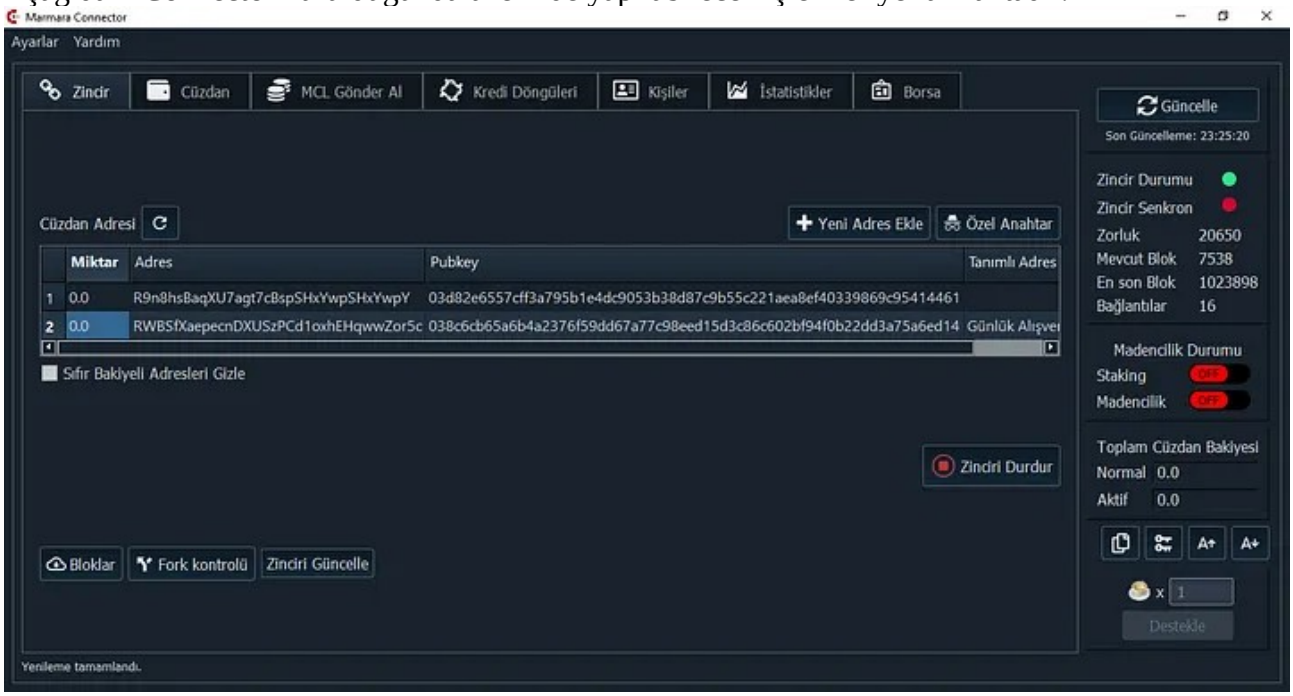
Marmara Connector’un kendisinin kurulması bu kadar basit.

Bknz [Releases · marmarachain/marmara-connector · GitHub]

MARMARA CONNECTOR NE YAPAR?

1. Marmara Connector noterlik yapmanız için gerekli Marmara Blokzinciri kurulumunu kolayca yapmanızı sağlar.
2. Alışveriş sırasında kredi döngüleriyle ilgili her türlü işlemleri yapmanızı sağlar.
3. Marmara Blokzinciri üzerinde gerekli diğer her türlü işlemleri, MCL koin cüzdanla ilgili işlemleri, bilgilendirme işlevlerini gerçekleştirir.

Aşağıdaki Connector kurulduğunda üzerinde yapılabilecek işlemler yer almaktadır.



Marmara Connector Kullanarak Marmara Blokzinciri Kurulumuyla İlgili Aşağıdaki linkten ilgili videoyu izleyebilirsiniz:
https://youtu.be/irb4y3iOsp8

BLOKZİNCİRİ SENKRONİZASYONU

Blokzinciri dağıtık noterliktir ve kaydın sadece bir yerde tutulduğu merkezi veritabanı sistemlerinin aksine her düğüm (noter) aynı verileri tutar. Bunun karşılığında ödül kazanmaya hak kazanır. O yüzden merkezi yerde kayıtların tutulduğu (örnek bankalar vs) binlerce kat daha güvenlidir. Tüm geçmiş veritabanı kayıtları tutulmadan zincir üzerinde noterlikle ilgili işlemleri vs yapamazsınız. Aşağıdaki linkteki videoda senkronizasyon işlemleri yer almaktadır.

https://youtu.be/n0TG93mXjBk

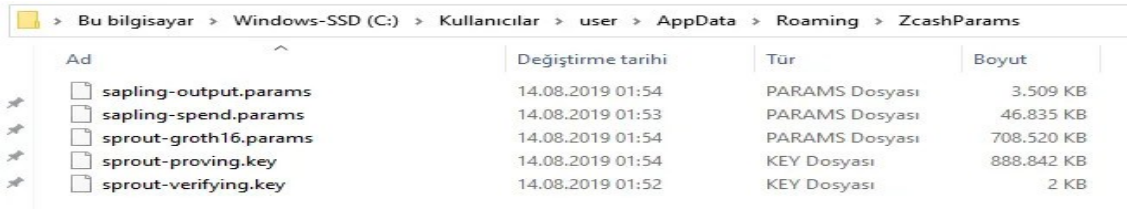
MARMARA CONNECTOR KURULUMU SIRASINDA OLASI SORUNLAR

Marmara Connector 3 şeyi yapar.

1. Marmara Blokzincirine çalıştırmak için gerekli dosyaları indirir. Bunu indirmek için curl programı kurulu olmalıdır. Bazı Windows sistemlerinde kurulu değilse elle kurmak gerekir.

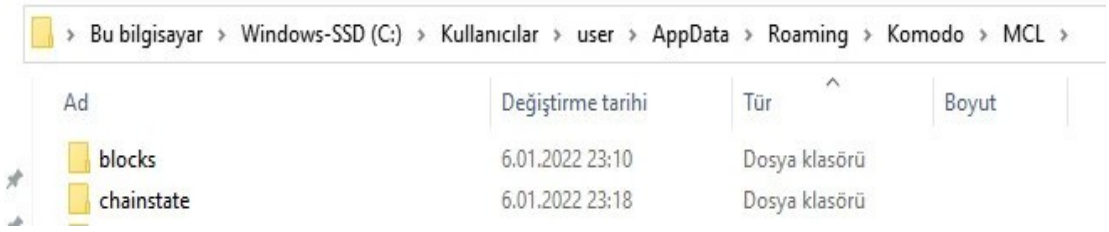
2. Marmara Blokzinciri işlemleri için Marmara ile ilgili aşağıdaki program dosyaları C:\Users\user\

AppData\Roaming\ZcashParams dizini altında hazır olmalıdır. Not: Aşağıdaki gibi dosyalara ve uzunluklarına dikkat ediniz.



3. Marmara Blokzinciri ile ilgili tüm işlemler için gerekli programlar C:\Users\user\marmara klasörü altına aşağıdaki gibi indirilmiş olmalıdır. Not: Aşağıdaki gibi dosyalara ve uzunluklarına dikkat ediniz.

4. Geçmiş blokzinciri kayıtlarına ait tüm veriler. Bunun en sağlıklı yolu canlı olarak indirilmesini beklemektir. Kurduktan sonra zinciri başlattığınızda Marmara Connector içinde otomatik indirilecektir. Tüm veriler indiğinde senkron oldu denmektedir. Hızlıca indirmek için günlük güncellenen Bootstrap dosyasını buradan indirerek hızlıca senkron olabilirsiniz. Marmara Connector bunu kendi içinde yapmaktadır. Bootstrap dosyası C:\Users\user\AppData\Roaming\Komodo\MCL dizininde MCL klasörü altına açılmalıdır. Senkron olan zinciri durdurduktan sonra aldığınız iki klasöre ait yedeği zincir verileri bozulduğunda kullanabilirsiniz.



Bootstrap https://explorer2.marmara.io/bootstrap adresinde günlük olarak yedeklenmektedir.

Alternatif link:
* https://eu.bootstrap.dexstats.info/MCL-bootstrap.tar.gz

Bootstrap için not: Bootstrap'te yalnızca blocks ve chainstate klasörü bulunur. İkisi yeterlidir. Zinciri durdurduktan sonra Wallet.dat dışındaki tüm mevcut klasörleri silin. Daha sonra bu iki klasörü açın. Hatta sisteminizde senkronizasyonu sağladığınızda Bootstrap'i indirmenize bile gerek kalmaz. Zaman zaman senkronize zinciri durdurun, blocks ve chainstate klasörlerini yedekleyin. İşte kendi Bootstrap dosyalarınız.

Aşağıdaki linklerden bilgi edinebilirsiniz:

Marmara Web Sitesi: https://marmara.io
Marmara Discord Grubu: https://marmara.io/discord
X (Twitter) Marmara: https://twitter.com/marmarachain
Marmara Credit Loops Youtube Kanalı: https://www.youtube.com/channel/UC_Ym-tICCd7ATlBU_1ToEKw/videos

****

Kaynaklar:

- https://marmarachain-mcl.medium.com/marmara-connector-ile-marmara-blokzincirinin-kolay-kurulumu-ve-mcl-koin-%C3%BCretimi-1a1e56fadfa7
- https://marmara.io/discord

Marmara Zinciri Kaynak Kodundan Derlemek

Temel Komodo yazılım paketi iki uygulama içerir.

komodod
komodod yazılım uygulaması, tüm Komodo blok zincirlerine güç veren Akıllı Zincir arka plan programıdır.

komodo-cli
komodo-cli yazılım uygulaması, geliştiricinin komut satırı aracılığıyla komodod'a API çağrıları yürütmesine olanak tanır.

Her ikisi de otomatik olarak kurulur
Bu yazılım uygulamalarının her ikisi de, aşağıdaki kurulum prosedürlerinden herhangi birinin parçası olarak ~/komodo/src/ dizinine yüklenir.

Kaynak Koddan Marmara Kredi Döngüleri Zeki Zincirini Derlemek
Marmara Kredi Döngüleri blokzinciri kaynak kodundan derlenebilir. Bu zorunlu bir işlem değildir (önceden derlenmiş hali de kullanılabilir), ancak kaynaktan derleme yöntemi kişinin en son yapılan yazılım yamalarına ve yükseltmelerine anında güncelleme yapmasına izin vermesi nedeniyle iyi bir uygulama olarak kabul edilmektedir.

Linux

Sistem Gereksinimleri

- Linux (Ubuntu gibi Debian tabanlı bir dağıtımla kurulum daha kolay gerçekleşir)
  Ubuntu için Ubuntu 20.04 LTS veya 22.04 LTS sürümleri önerilir.
- 64-bit İşlemci
- Minimum 2 çekirdekli CPU (4 çekirdekli tavsiye edilir)
- Minimum 4GB RAM

Başlangıç
Sisteminizin güncel olduğunu doğrulayın.

```
sudo apt-get update
sudo apt-get upgrade -y
```

Bağımlılık Paketlerini Yükleyin

```
sudo apt-get install build-essential pkg-config libc6-dev m4 g++-multilib autoconf libtool ncurses-dev unzip git python python3-zmq zlib1g-dev wget curl bsdmainutils automake cmake clang ntp ntpdate nano -y
```

Bu işlem İnternet bağlantınıza bağlı olarak biraz zaman alır. İşlemin arka planda çalışmasına izin verin. Tamamlandıktan sonra aşağıdaki adımları takip ederek Marmara Kredi Döngüleri zeki zincirini kurun.

Marmara Kredi Döngüleri Deposunu Klonlayın

```
cd ~
git clone https://github.com/marmarachain/marmara --branch master --single-branch
```

Kullandığınız sistemde 8 GB veya daha fazla RAM varsa aşağıdaki adımları atlayıp, Zcash Parametrelerinin Çekilmesi adımına geçebilirsiniz.

Takas Alanı Boyutunu 4 GB Olarak Değiştirmek

Takas Alanı (Swap), fiziksel RAM belleğinin miktarı dolduğunda kullanılan, diskteki bir alandır. Sistemde 4 GB RAM varsa, RAM'in tükenmesi durumlarını önlemek için takas alanının yapılandırılması gerekir.

Swap alanının olup olmadığını `sudo swapon --show` komutu ile kontrol edebilirsiniz. Sistemde swapfile mevcut olup en az 4 GB olacak şekilde ayarlanmadıysa bu durumda öncelikli olarak `sudo swapoff /swapfile` komutuyla swap'ı kapatınız.

Aşağıdaki komutları kullanarak 4 GB RAM takas alanı oluşturabilirsiniz.

```
sudo fallocate -l 4G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
```

Zcash Parametrelerinin Çekilmesi

```
cd marmara
./zcutil/fetch-params.sh
./zcutil/build.sh -j$(nproc)
```

Bu adım tamamlandıktan sonra, komodod ve komodo-cli dosyalarını src klasörünün içerisinde bulunacaktır.

Aşağıdaki adımların izlenmesi isteğe bağlıdır.

Swappiness Ayarlaması

Swappiness linux çekirdeğinin RAM içeriğini takas için ne kadar (ve ne sıklıkta) kopyaladığını tanımlayan çekirdek parametresidir. Aşağıdaki komutta parametrenin değeri “10” olarak ayarlanır. Takas parametresinin değeri ne kadar yüksek olursa, Linux çekirdeğinin o kadar agresif bir şekilde değişeceğini unutmayın.

```
sudo sysctl vm.swappiness=10
```

Bu ayar bir sonraki yeniden başlatmaya kadar geçerli olacaktır. /etc/sysctl.conf dosyasına şu satırı ekleyerek yeniden başlatma sırasında bu değeri otomatik olarak ayarlayabiliriz:
`sudo nano /etc/sysctl.conf vm.swappiness=10`

UFW'yi etkinleştirme

Bu adım MCL'yi yüklemek için gerekli değildir ancak sunucu güvenliği amacıyla kullanılabilir.

- UFW'nin durumunu şu komutla kontrol edin:

```
sudo ufw status
```

- UFW paketi aşağıdaki komut çalıştırılarak kurulabilir:

```
sudo apt install ufw
```

- Aşağıdaki komutları yürüterek UFW'yi etkinleştirin ve bağlantılara izin verin:

```
echo y | sudo ufw enable
sudo ufw allow ssh
sudo ufw allow "OpenSSH"
sudo ufw allow 33824
sudo ufw allow out 33824
```

MacOS

Sistem gereksinimleri

- OSX (Sürüm > 10.11)
- Minimum 4GB RAM (8GB+ önerilir)

Komut satırı araçlarının kurulu olduğundan emin olun

Bir terminalde aşağıdaki komutu verin:

```
xcode-select –install
```

Brew'in kurulu olduğundan emin olun

Bağımlılıkları yüklemek için brew yazılımını kullanıyoruz. Brew'in en son sürümü zaten kuruluysa bu adımı atlayabilirsiniz.

```
usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew install/master/install)"
```

Bağımlılıkları Yüklemek için Brew Kullanın

Her komutu ayrı ayrı yürütün

```
brew update
brew upgrade
brew tap discoteq/discoteq; brew install flock
brew install autoconf autogen automake
brew install gcc@8
brew install binutils
brew install protobuf
brew install coreutils
brew install wget
```

Marmara Kredi Döngüleri Deposunu Klonlayın

```
cd ~
git clone https://github.com/marmarachain/marmara --branch master --single-branch
```

Zcash Parametrelerinin Çekilmesi

```
cd marmara
./zcutil/fetch-params.sh
./zcutil/build-mac.sh -j8
```

Bu adım tamamlandıktan sonra komodod & komodo-cli dosyalarını src dizininde bulacaksınız. Bunları tamamladıktan sonra buradaki talimatları takip ederek Marmara Zincirini başlatın.

Windows

Windows yazılımı doğrudan bir Windows işletim sisteminde derlenemez. Bunun yerine, yazılımın bir Linux makinesinde derlenmesi gerekir (Ubuntu önerilir) ve ilgili .exe çıktısı daha sonra Windows işletim sistemine taşınmalıdır.

Sistem Gereksinimleri

- Linux işletim sistemi veya Ubuntu Linux'ün Sanal Makine tabanlı kurulumu
- Linux sistem gereksinimlerini burada bulabilirsiniz.

Bağımlılık paketlerini yükleyin

```
sudo apt-get install build-essential pkg-config libc6-dev m4 g++-multilib autoconf libtool ncurses-dev unzip git python python-zmq zlib1g-dev wget libcurl4-gnutls-dev bsdmainutils automake curl libsodium-dev cmake mingw-w64
```

Rust Kurulumu

```
curl https://sh.rustup.rs -sSf | sh
source $HOME/.cargo/env
rustup target add x86_64-pc-windows-gnu
```

Derleyiciyi POSIX İş Parçacığı Modelini Kullanacak Şekilde Yapılandırma

Aşağıdaki komutu çalıştırın:

```
sudo update-alternatives --config x86_64-w64-mingw32-gcc
```

Yukarıdaki komutu verdikten sonra aşağıda belirtildiği gibi POSIX seçeneğini seçin:

```
Selection    Path                                            Priority   Status
------------------------------------------------------------------------------------------------------------
  0          /usr/bin/x86_64-w64-mingw32-gcc-win32           60         auto mode
  1          /usr/bin/x86_64-w64-mingw32-gcc-posix           30         manual mode
* 2          /usr/bin/x86_64-w64-mingw32-gcc-win32           60         manual mode

Press <enter> to keep the current choice[*], or type selection number: 1
```

Aşağıdaki komutu verin:

```
sudo update-alternatives --config x86_64-w64-mingw32-g++
```

Yukarıdaki komutu yürüttükten sonra aşağıdaki gibi POSIX seçeneğini seçin:

```
There are 2 choices for the alternative x86_64-w64-mingw32-g++ (providing /usr/bin/x86_64-w64-mingw32-g++).

  Selection    Path                                        Priority   Status
------------------------------------------------------------------------------------------------
  0            /usr/bin/x86_64-w64-mingw32-g++-win32      60        auto mode
  1            /usr/bin/x86_64-w64-mingw32-g++-posix      30        manual mode
* 2            /usr/bin/x86_64-w64-mingw32-g++-win32      60        manual mode

Press <enter> to keep the current choice[*], or type selection number: 1
```

Marmara Deposunu klonlayın

```
git clone https://github.com/marmarachain/marmara --branch master --single-branch
cd marmara
```

Zcash Parametrelerinin Çekilmesi

```
./zcutil/fetch-params.sh
./zcutil/build-win.sh -j$(nproc)
```

Bu adım tamamlandıktan sonra src dizininde komodod.exe ve komodo-cli.exe dosyalarını bulacaksınız.

Bu dosyaları kopyalayıp Windows makinesine taşıyın. Buradan devam edin: https://github.com/marmarachain/marmara/wiki/Install-Marmara-from-Precompiled-Binaries#windows

Marmara Blokzincirine Başlangıç

Marmara Blokzincirine nasıl başlanır?
Temel Komodo yazılım paketi iki uygulama içerir.

komodod
komodod yazılım uygulaması, tüm Komodo blok zincirlerine güç veren Akıllı Zincir arka plan programıdır.

komodo-cli
komodo-cli yazılım uygulaması, geliştiricinin komut satırı aracılığıyla komodod'a API çağrıları yürütmesine olanak tanır.

MacOS

komodod.exe ve komodo-cli.exe dosyalarının olduğu dizini bulun. Komut satırını açın ve bu dizine geçin.

komodo-cli ve getinfo kullanımı
MCL Smartchain hakkında genel bilgileri görmek için aşağıdaki gibi getinfo komutunu çalıştırın:

```
./komodo-cli -ac_name=MCL getinfo
```

Blokların ve en uzun zincirin döndürülen özellikleri birbirine eşit olduğunda, arka plan programı ağla eşitlemeyi tamamlar.

Linux işletim sisteminde olduğu gibi MARMARA komutlarını yayınlayarak Mac OS kullanırken MARMARA blok zinciri ile etkileşim kurulabilir.

Windows

komodod.exe ve komodo-cli.exe dosyalarının olduğu dizini bulun. Komut satırını açın ve bu dizine geçin.

Örneğin, ilgili dosyaların Masaüstünde MCL klasörü altında bulunduğu göz önüne alındığında; komut aşağıdaki gibi olmalıdır:

```
cd Desktop\MCL\
```

komodo-cli ve getinfo kullanımı

Aşağıda getinfo komutu için bir örnek verilmiştir:

```
komodo-cli -ac_name=MCL getinfo
```

Linux

Marmara zinciri kaynak koddan indirip kurduysanız, komutları komodo/src dizini altında çalıştırabilirsiniz. Yapılandırmanıza göre komodod ve komodo-cli farklı dizinler altında olabilir. Bu nedenle, nerede olduklarını bulun ve komut satırında olduğunuz dizini cd komutu ile değiştirerek bulundukları dizine gidin.

Marmara Zincirini aşağıdaki parametrelerle çalıştırın:

```
./komodod -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -ac_marmara=1 -ac_staked=75 -ac_reward=3000000000 &
```

Bağlanana ve senkronize olana kadar bekleyin. Aşağıdakileri gerçekleştirerek düğümün zincire senkronize edilip edilmediğini kontrol edebilirsiniz:

```
./komodo-cli -ac_name=MCL getinfo
```

İndeksleme (Bootstrap - Tüm blokları indirme işlemini sabitlemek için isteğe bağlı adım)**

Yeni gelenlerin tüm blokların makinelerine indirilmesini beklemeleri gerekir. Bu işlemi hızlandırmak için bootstrap indirilebilir ve kullanılabilir.

Aşağıdaki komutu uygulayarak Marmara blockchain'i durdurun:

Linux:

```
./komodo-cli -ac_name=MCL stop
```

Windows:

```
komodo-cli.exe -ac_name=MCL stop
```

Bootstrap'i komut satırından yüklemek için aşağıdaki komutu çalıştırın:

Linux:

```
wget https://eu.bootstrap.dexstats.info/MCL-bootstrap.tar.gz
```

Şimdi, aşağıdaki tar komutu önyükleme (bootstrap) içeriğini aşağıda gösterildiği gibi belirli bir dizine çıkartacaktır:

```
tar -xvf MCL-bootstrap.tar.gz -C .komodo/MCL
```

Şimdi aşağıdaki komutu kullanarak Marmara Zincirini yeniden başlatın:

Linux:

```
./komodod -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -ac_marmara=1 -ac_staked=75 -ac_reward=3000000000 &
```

Windows:

```
komodod.exe -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -ac_marmara=1 -ac_staked=75 -ac_reward=3000000000 &
```

** Bootstrap dosyası bozulabileceğinden, blokları indirmek için Bootstrap kullanmak önerilen bir yaklaşım değildir. Önerilen yaklaşım, Marmara blockchain'i en baştan indekslemektir.**

Pubkey Oluşturma ve Pubkey ile MCL Başlatma

Marmara Kredi Döngüleri'ni kullanabilmek için kullanıcının bir pubkey'inin olması ve zinciri bir pubkey ile başlatması gerekir. Aksi takdirde, çıkarılan veya yatırılan tüm MCL'ler pubkey adresine gönderildiğinden, zeki zincirdeki herhangi bir madencilik veya staking boşuna olacaktır.

Pubkey almak için Marmara Zinciri normal başlatma parametreleriyle çalıştırın ve getnewaddress API komutunu çalıştırın:

Linux:

```
./komodo-cli -ac_name=MCL getnewaddress
```

Windows:

```
komodo-cli.exe -ac_name=MCL getnewaddress
```

getnewaddress komutu sırayla yeni bir adres döndürür:

```
DO_NOT_USE_THIS_ADDRESSgg5jonaes1J5L786
```

Şimdi validateaddress komutunu çalıştırarak adresinizi doğrulayın.

Linux:

```
./komodo-cli -ac_name=MCL validateaddress
DO_NOT_USE_THIS_ADDRESSgg5jonaes1J5L786
```

Windows:

```
komodo-cli.exe -ac_name=MCL validateaddress
DO_NOT_USE_THIS_ADDRESSgg5jonaes1J5L786
```

Bu, birçok özelliğe sahip bir json nesnesi döndürecektir. Özelliklerde biri görebilir:

```
"pubkey":
"DO_NOT_USE_THIS_ADDRESS019n79b0921a1be6d3ca6f9e8a050mar17eb845fe46b9d756"
```

Bu sizin MCL pubkey'iniz olacak, not aldığınızdan emin olun. Şimdi bunu daemon'a belirtmelisiniz. Bunu yapmak için önce aşağıdaki komutu kullanarak arka planı durdurun:

Linux:

```
./komodo-cli -ac_name=MCL stop
```

Windows:

```
komodo-cli.exe -ac_name=MCL stop
```

MCL'yi Pubkey ile Başlatma

Gerekli parametreleri kullanarak arka plan programınızı yeniden başlatın ve pubkey'inizi ek bir parametre olarak eklediğinizden emin olun. Örneğin:

Linux:

```
./komodod -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -
addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -
```

```
ac_marmara=1 -ac_staked=75 -ac_reward=3000000000 -gen -genproclimit=1 -
pubkey=DO_NOT_USE_THIS_ADDRESS019n79b0921a1be6d3ca6f9e8a050mar17eb845fe46b9
d756 &
```

Windows:

```
komodod.exe -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -
addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -
ac_marmara=1 -ac_staked=75 -ac_reward=3000000000 -gen -genproclimit=1 -
pubkey=DO_NOT_USE_THIS_ADDRESS019n79b0921a1be6d3ca6f9e8a050mar17eb845fe46b9
d756 &
```

-genproclimit madencilik için kullanılacak iş parçacığı sayısını ayarlar.

Yukarıdaki komutta -genproclimit parametresi gen -genproclimit=1 gibi 1 olarak ayarlanmışsa Marmara Zincirini tek (1) CPU ile başlatır. Bunu 3 gibi herhangi bir sayıda çekirdeğe ayarlayabileceğinizi unutmayın.

Yukarıdaki komutta -genproclimit parametresi -1 olarak ayarlanmışsa, örneğin -gen -genproclimit=-1 tüm CPU ile Marmara Zincirini başlatır.

Yukarıdaki komutta -genproclimit parametresi 0 olarak ayarlanmışsa, örneğin -gen -genproclimit=0, Marmara Zincirini "staking" modu açık olarak başlatarak aktifteki ve döngülerdeki MCL'ler ile staking yapar.

dumpprivkey

dumpprivkey yöntemi, belirtilen adrese karşılık gelen özel anahtarı gösterir. Bunun için komut demo amaçlı aşağıda verilmiştir:

Linux:

```
./komodo-cli -ac_name=MCL dumpprivkey "PWqwYaWNEVT7V9SdfFHARWnoB7vcpSfdvs"
```

Windows:

```
komodo-cli.exe -ac_name=MCL dumpprivkey "PWqwYaWNEVT7V9SdfFHARWnoB7vcpSfdvs"
```

Komutun yanıtı, aşağıdaki örnekte olduğu gibi özel adresi verir:

```
DONOTUSETHISxxxxxxxxxxxxxxxxxx7KkCmRnnSg7iXvRUqhYoxC9Y
```

importprivkey

importprivkey yöntemi, cüzdanınıza özel bir anahtar (dumpprivkey tarafından döndürüldüğü gibi) ekler. Komut, burada açıklandığı gibi birkaç argüman alabilir. Bunun için en basit komut biçimi aşağıda demo amaçlı verilmiştir.

Linux:

```
./komodo-cli -ac_name=MCL importprivkey
"DONOTUSETHISxxxxxxxxxxxxxxxxx7KkCmRnnSg7iXvRUqhYoxC9Y"
```

Windows:

```
komodo-cli.exe -ac_name=MCL importprivkey
"DONOTUSETHISxxxxxxxxxxxxxxxxx7KkCmRnnSg7iXvRUqhYoxC9Y"
```

Komutun yanıtı, aşağıdaki demo yanıtında olduğu gibi cüzdan adresini verir:

```
R9z796AehK5b6NCPeVkGUHSpJnawerf8oP
```

Şimdi, MCL pubkey'i almak için cüzdan adresi validateaddress aracılığıyla doğrulanabilir:

Linux:

```
./komodo-cli -ac_name=MCL validateaddress R9z796AehK5b6NCPeVkGUHSpJnawerf8oP
```

Windows:

```
komodo-cli.exe -ac_name=MCL validateaddress R9z796AehK5b6NCPeVkGUHSpJnawerf8oP
```

Yukarıdaki komutun cevabından ilgili MCL pubkey'i aldıktan sonra, Marmara Zincirini buradaki gibi o pubkey ile yeniden başlatın.

Marmara Blockchain'de Kendi Kelimelerinizle Cüzdan Adresi Oluşturulması

Linux işletim sistemi için; daha önce açıklandığı gibi, bir cüzdan adresi almak için `./komodo-cli -ac_name=MCL getnewaddress` komutu ve ardından ilgili Özel Anahtarı (Privkey) almak için `./komodo-cli -ac_name=MCL validateaddress "Rwalletno"` komutu kullanılabilir.

Ancak bu şekilde alınan Özel Anahtar (Privkey), hatırlanması zor bir harf ve sayı kombinasyonundan oluşur. Bu özel anahtarı yalnızca sizin tarafınızdan güvenli ve ulaşılabilir tutmanın, varlıklarınıza ulaşmanın çok önemli ve önemli bir yolu olduğunu unutmayın. Bu nedenle, kişi kendi kelime grubundan bir özel anahtar oluşturabilir ve gerektiğinde ilgili özel anahtarı oluşturmak için bu kelimelerin kaydını tutabilir.

Bu amaçla, içinde özel bir karakter bulunmayan 12 adet kelimeden oluşan bir dizi oluşturun. Örneğin: "can create your own keywords and generate privkey from these whenever needed".

Ardından, ilgili özel anahtarın oluşturulmasını sağlamak için aşağıdaki komutu kullanın:

Linux:

```
./komodo-cli -ac_name=MCL convertpassphrase "can create your own keywords and generate
privkey from these whenever needed"
```

Windows:

```
komodo-cli.exe -ac_name=MCL convertpassphrase "can create your own keywords and generate
privkey from these whenever needed"
```

Komut sırayla aşağıdaki JSON Nesnesini döndürür:

```
{
  "agamapassphrase": "can create your own keywords and generate privkey from these whenever
needed",
  "address": "RCna416S62uNVAKMwRuFtmRH2X3xJuTidD",
```

```
  "pubkey": "03a3c416d3bafe44448dcb3a56e5dd13db5cc3e6992051a61a2f5675b4796c0d7a",
  "privkey": "9005f5b2593ab3e662e1330ceca848d62fb2021165c6484ac7037cb4efc5317e",
  "wif": "UtqWJyhXjJWYigKyKCm5ML96LvSZFDfZrT2HYXAdGs8byZ3uCMzy"
}
```

Daha sonra, ilgili özel anahtarı cüzdan adresine eklemek için importprivkey yöntemi kullanılabilir:

Linux:

```
./komodo-cli -ac_name=MCL importprivkey
"UtqWJyhXjJWYigKyKCm5ML96LvSZFDfZrT2HYXAdGs8byZ3uCMzy"
```

Windows:

```
komodo-cli.exe -ac_name=MCL importprivkey
"UtqWJyhXjJWYigKyKCm5ML96LvSZFDfZrT2HYXAdGs8byZ3uCMzy"
```

Şimdi, varlıkların sahibinin, gerektiğinde ilgili özel anahtarı oluşturmak için anahtar kelime kombinasyonunun kaydını güvenli bir şekilde tutması gerekiyor. Özel anahtarların her zaman gizli tutulması gerektiğini ve anahtar kelimelerin de gizli tutulması gerektiğini unutmayın!

Cüzdanın Yedeğini Almak

Wallet.dat dosyasının yedeklenmesi, bir kişinin varlıklarını barındırdığı için çok önemlidir.

Linux

Bir linux makinesinde dosya şurada bulunabilir: ~/.komodo/MCL/wallet.dat

Bu dosyayı yedeklemenin bir yöntemi, dosyanın bir kopyasını arşivlemektir.

```
#Copy the wallet.dat file
cp -av ~/.komodo/MCL/wallet.dat ~/wallet.dat

#Rename the wallet.dat file
mv ~/wallet.dat ~/2020-08-09-wallet_backup.dat

# Make an archieve of the wallet.dat file
tar -czvf ~/2020-08-09-wallet_backup.dat.tgz ~/2020-08-09-wallet_backup.dat

# Move the final file to a secure location
```

Marmara Chain'deki düğümünüzün Staking/Madencilik modunu kontrol etme

Aşağıdaki komut düğümün modunu kontrol etmeye yardımcı olur:

Linux:

```
./komodo-cli -ac_name=MCL getgenerate
```

Windows:

```
komodo-cli.exe -ac_name=MCL getgenerate
```

Yukarıda verilen komut yürütüldüğünde aşağıdaki JSON nesnesi döndürülür:

```
{
```

```
  "staking": false,
  "generate": true,
  "numthreads": 1
}
```

Yukarıda düğümün madencilik düğümünde olduğu görülebilir.

"staking": false staking kapalı anlamındadır.

"generate": true madencilik aktif anlamındadır.

"numthreads": 1 madencilik için kullanılan çekirdek sayısını ifade eder. Bu durumda bu parametre daha önceden bir değere ayarlanmıştır.

Düğümün modunu stake etme olarak değiştirmek için aşağıdaki komutu yürütün:

Linux:
```
./komodo-cli -ac_name=MCL setgenerate true 0
```

Windows:
```
komodo-cli.exe -ac_name=MCL setgenerate true 0
```

Şimdi düğümün durumu kontrol ediliyor:

Linux:
```
./komodo-cli -ac_name=MCL getgenerate
```

Windows:
```
komodo-cli.exe -ac_name=MCL getgenerate
```

Bu komut bir JSON nesnesi döndürür:
```
{
  "staking": true,
  "generate": false,
  "numthreads": 0
}
```

Unutmayın lütfen "staking": true aktif MCL'niz yoksa kullanılmayacaktır!

Bir Adrese Coin Göndermek

Aşağıdaki komutu vererek doğrudan belirli bir adrese ödeme gönderebilirsiniz. Ödeme, ilgili cüzdanda mevcut olan Normal Tutar temel alınarak yapılır. Tutar en yakın 0,00000001'e yuvarlanır. Yapılan işlem karşılığında kişinin Normal Tutarından işlem ücreti kesilir.

Linux:

```
./komodo-cli -ac_name=MCL sendtoaddress "MCL_address" amount
```

Windows:

```
komodo-cli.exe -ac_name=MCL sendtoaddress "MCL_address" amount
```

> **Not: Bu komut, ödemeyi doğrudan belirtilen adrese gönderir ve dikkatli kullanılmalıdır. Çünkü bu yöntemle gönderilen ödemenin daha sonra geri alınması mümkün değildir.**

Marmara Zinciri'nde Node ve Cüzdan ile İlgili Detaylara Ulaşım

Aşağıdaki komutlar, düğümünüz ve cüzdanınız hakkında ayrıntılı bilgi almak için kullanışlıdır.

- getinfo

Linux:

```
./komodo-cli -ac_name=MCL getinfo
```

Windows:

```
komodo-cli.exe -ac_name=MCL getinfo
```

Aşağıda tipik bir sonuç verilmiştir. **”synced”:false** (senkronize değil) olduğunu unutmayın:

```
{
  "version": 3000300,
  "protocolversion": 170009,
  "KMDversion": "0.5.3",
  "synced": false,
  "notarized": 0,
  "prevMoMheight": 0,
  "notarizedhash":
"0000000000000000000000000000000000000000000000000000000000000000",
  "notarizedtxid":
"0000000000000000000000000000000000000000000000000000000000000000",
  "notarizedtxid_height": "mempool",
  "KMDnotarized_height": 0,
  "notarized_confirms": 0,
  "walletversion": 60000,
  "balance": 0.00000000,
  "blocks": 72200,
  "longestchain": 586375,
  "tiptime": 1583613252,
  "difficulty": 38227.25456170298,
  "keypoololdest": 1614160564,
  "keypoolsize": 101,
  "paytxfee": 0.00000000,
  "sapling": 61,
```

```
  "timeoffset": 0,
  "connections": 16,
  "proxy": "",
  "testnet": false,
  "relayfee": 0.00000100,
  "errors": "",
  "CCid": 2,
  "name": "MCL",
  "p2pport": 33824,
  "rpcport": 33825,
  "magic": -1687041972,
  "premine": 2000000,
  "reward": "3000000000",
  "halving": "0",
  "decay": "0",
  "endsubsidy": "0",
  "notarypay": "0",
  "staked": 75
}
```

getinfo komutu bazı önemli ayrıntıları döndürür, "version" aracılığıyla MARMARA'nın sürümü; düğümünüzün “synced” yoluyla senkronizasyon durumu (bu parametrenin değeri, "blocks" ve "longestchain" parametreleri eşitse doğrudur); zincirin "difficulty" yoluyla zorluğu; "connections" yoluyla zincire en yakın bağlı düğümlerin sayısı; "pubkey" aracılığıyla zincire bağlandığınız pubkey gibi.

- getpeerinfo

getpeerinfo komutu, düğümünüzün etrafındaki zincire en yakın bağlı düğümler hakkında ayrıntılı bilgi döndürür.

Linux:

```
./komodo-cli -ac_name=MCL getpeerinfo
```

Windows:

```
komodo-cli.exe -ac_name=MCL getpeerinfo
```

- marmarainfo

Linux:

```
./komodo-cli -ac_name=MCL marmarainfo 0 0 0 0 pubkey
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmarainfo 0 0 0 0 pubkey
```

marmarainfo komutu, bazı önemli ayrıntıları döndürür, "myPubkeyNormalAmount" aracılığıyla pubkey'deki normal miktar; "myActivatedAmount" aracılığıyla etkinleştirilen tutar; "Loops" yoluyla yapılan kredi döngülerinin ayrıntıları; "TotalLockedInLoop" aracılığıyla kredi döngülerinde

kilitlenen toplam tutar; "numclosed" aracılığıyla kapatılan kredi döngülerinin sayısı; ve "closed" ile kapatılan kredi döngülerinin ayrıntıları gibi.

MCL'leri Aktifleştirmek ve Aktifleştirilenlerin Kilitlerini açmak

- marmaralock

MCL'leri aktifleştirmek yani kilitlemek için marmaralock kullanılır. Staking için aktif MCL'lere ihtiyaç vardır ve eğer yoksa staking modu açık olsa bile staking yoluyla hiçbir blok bulunamaz. Komut örneği aşağıda verilmiştir ve amount (miktar) 1000 gibi rakamsal bir MCL miktarıyla değiştirilecektir.

Linux:

```
./komodo-cli -ac_name=MCL marmaralock amount
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaralock amount
```

Bu komut, işlemin sonucunu ve bunun için oluşturulan hex'i içeren bir JSON nesnesi döndürür. Aşağıda verilen "hex"in gösterim amaçlı olduğunu unutmayın.

```
{
  "result": "success",
  "hex":
"0400008085202f89020039b219200ae4b5c83d77bffce7a8af054d6fb.....e9181f6aac3e1beb1e260e9a1f49ed24e6ac00000000edeb040000000000000000000000000"
}
```

Şimdi bu işlemi onaylamak için JSON nesnesi üzerinden dönen hex'i kopyalayın ve aşağıda verilen sendrawtrasaction komutu ile doğrulayın:

Linux:

```
./komodo-cli -ac_name=MCL sendrawtransaction
0400008085202f89020039b219200ae4b5c83d77bffce7a8af054d6fb.....e9181f6aac3e1beb1e260e9a1f49ed24e6ac00000000edeb040000000000000000000000000
```

Windows:

```
komodo-cli.exe -ac_name=MCL sendrawtransaction
0400008085202f89020039b219200ae4b5c83d77bffce7a8af054d6fb.....e9181f6aac3e1beb1e260e9a1f49ed24e6ac00000000edeb040000000000000000000000000
```

Yukarıdaki komut blok zincirinde başarıyla yürütülürse yanıt olarak bir işlem kimliği (transaction id) verir. Bu işlemin doğrulanıp doğrulanmadığı, ilgili id'nin Marmara Explorer sitesinde aranması ile kontrol edilebilir. Aktifleştirilmiş koinleri görmek için daha önce verilen marmarainfo komutunu kullanın ve "myActivatedAmount" parametresindeki değeri arayın. Ham işlemlerin (raw transactions) bellek havuzunda (mempool) toplandığını ve bloğa kaydedilen işlemi görmek için birkaç bloğun bulunması gerekebileceğini unutmayın.

- marmaraunlock

marmaraunlock, MCL'lerin kilidini açmak yani normal miktara çevirmek için kullanılır. Normal tutar (Normal amount), daha önce açıklanan sendtoaddress komutu aracılığıyla ödemelerin doğrudan bir adrese gönderilmesi için kullanılır. Aşağıdaki komutlardaki kilidi açılacak amount (tutar) 500 gibi rakamsal bir MCL miktarıyla değiştirilmelidir.

Linux:
```
./komodo-cli -ac_name=MCL marmaraunlock amount
```

Windows:
```
komodo-cli.exe -ac_name=MCL marmaraunlock amount
```

Daha önce de açıkladığımız gibi bu işlemin yukarıda verilen sendrawtrasaction komutu ile doğrulanması gerekmektedir. Bunun için marmaraunlock komutunun döndürdüğü hex'i kopyalayıp, sendrawtrasaction komutuyla birlikte kullanın.

- listaddressgroupings

listaddressgroupings is used to list the pairs of wallet addresses and respective normal amounts in them. The usage is given in the command below.

Linux:
```
./komodo-cli -ac_name=MCL listaddressgroupings
```

Windows:
```
komodo-cli.exe -ac_name=MCL listaddressgroupings
```

- Marmaraamountstat

Bu komut, sahibinden bağımsız olarak Marmara blok zincirinin tüm aktif, normal ve kilitli döngü miktarını almak için kullanılır. Begin_height ve end_height girişleri sıfır(0) olarak ayarlanırsa bloklardan tüm veri setini alır.

Linux:
```
./komodo-cli -ac_name=MCL marmaraamountstat begin_height end_height
```

Windows:
```
komodo-cli.exe -ac_name=MCL marmaraamountstat begin_height end_height
```

Linux OS için örnek:
```
./komodo-cli -ac_name=MCL marmaraamountstat 50000 60000
```

Örnek Sonuç:
```
{
  "result": "success",
```

```
  "BeginHeight": 50000,
  "EndHeight": 60000,
  "TotalNormals": 166854.85464948,
  "TotalPayToScriptHash": 0.00000000,
  "TotalActivated": 582681.11953150,
  "TotalLockedInLoops": 237510.00000000,
  "TotalUnknownCC": 0.02820000,
  "SpentNormals": 161377.88109325,
  "SpentPayToScriptHash": 0.00000000,
  "SpentActivated": 382128.12128773,
  "SpentLockedInLoops": 143510.00000000,
  "SpentUnknownCC": 0.00000000
}
```


help
Bu komut, Marmara zeki zincirine ilişkin tüm yöntemleri almak için kullanılır.

Linux:
```
./komodo-cli -ac_name=MCL help
```

Windows:
```
komodo-cli.exe -ac_name=MCL help
```

Belirtilen yöntem adıyla aşağıdaki komutu vererek belirli bir yöntemin parametrelerini de kontrol edebilirsiniz. Örneğin:

Linux:
```
./komodo-cli -ac_name=MCL help getinfo
```

Windows:
```
komodo-cli.exe -ac_name=MCL marmaraamountstat
```

****

Kaynak: https://github.com/marmarachain/marmara/wiki/Getting-Started-with-Marmara

Referanslar
Komodo Antara Framework ve ayrıntıları hakkında daha ayrıntılı bilgi için lütfen genişletilmiş geliştirici belgelerine bakın:
https://developers.komodoplatform.com/

Marmara Kredi Döngüleri Nasıl Yapılır?

Mevcut Marmara Kredi döngüleri şu anda %100 teminatlandırma modunda olan Protokol 1'e dayalı olarak çalışmaktadır. % 100 teminatlandırma keşideci tarafından hem kendisi hem de hamil adına yapılır. Hem keşideci hem de hamil, blockzincir ödülleri almak için 3 kat stake etme şansına sahip olur. Keşideci, kredinin vade tarihine kadar 3 kat staking şansına sahipken, hamil, krediyi keşideci ile stake etmeye devam edecek yeni bir hamile ciro edene/transfer edene kadar 3 kat staking şansına sahiptir. Kredi döngüleri yalnızca aktifleştirilmiş MCL'ler kullanılarak yapılabilir.

Terminoloji

Keşideci: Kredi döngüsünde ilk olarak kredi oluşturan kişi. Kredi döngüsünde ilk düğümü oluşturan kişidir. Kredi, vade tarihine kadar %100 veya sıfır teminatla teminatlandırılabilir.

Hamil (Sahip): Kredi döngüsündeki son düğüme her zaman hamil (sahip) denir. Birisi bir döngü içinde kredi aktardığında, bu düğüm anında ciranta haline gelir.

Ciranta: Kredi döngüsünün ilk düğümü olan keşideci ile son düğüm arasına giren ve krediyi bir sonraki düğüme aktaran diğer tüm düğümler.

Vade: Mevcut bloktan başlayarak kredinin vadesinin kaç blok sonra bittiğini göstermektedir. Vade blokların çıkışına göre hesaplanır. Marmara zincirinde bir günde çıkan blok sayısı 1440'tır (Bir Saatte 60 blok çarpı 1 günde 24 saat sonucu). 100 günlük bir kredi için, Vade 1440x100 = 144,000 blok şeklinde hesaplanır.

Mahsuplaşma (Settlement): Bir kredinin vadesi dolduğunda, ödeme döngüde son düğüm olan hamile aktarılır. Vade tamamlanma durumunda mahsuplaşma otomatik veya manuel olarak yapılır.

Eskrov (Aracı): Protokol 2 için güvene dayalı taraflar. EscrowOn false ise, %100 teminatlandırma kullanılır ve uzlaşma otomatik olur. %100 tam güvenli sürüm olarak çalışan protokol 1'de eskrova gerek yoktur.

Avalist: Avalistler, ilave ortak olarak MCL ile kredi çıkaran veya onaylayanları destekler ve bu destekle 3 kat staking kazanabilir. İtfa edilmemesi durumunda fonları kullanılır. Avalistler yalnızca Protokol 2'de mevcuttur. Parametre avalcount, protokol 1 için her zaman sıfırdır.

Blokaj Miktarı (BlockageAmount): Bu terim Versiyon 2'ye ait bir parametredir. Ciranta keşideciden karşılıksızlığa karşı teminat sağlaması için bir miktar koymasını isteyebilir. Bu durumda, Keşideci ortaya koyduğu teminata karşılık 3 kat kadar staking ile kazanma şansı elde eder.

Zaman Aşımı (Dispute Expiry): Versiyon 2'de kredi döngülerinde olası bir karşılıksızlık problemini çözmede verilen süreye denir. Bu tarihe kadar mashuplaşmanın (settlement) yapılmış olması gerekir. Bu tarihten sonra hamil hak iddia edemez. Bir keşideci, teminatlandırma olmadan veya yetersiz teminatlandırma ile protokol 2 kapsamında bir kredi oluştururken bu süreyi bloklar halinde kullanabilir. Bu süre sona ermeden önce, bir eskrov, ödeme yapılmamasını çözmek için keşideci ile yapılan anlaşmaya göre tüm işlemleri yapmalıdır. Aksi takdirde eskrov sistemde cezalandırılır.

Marmara Kredi Döngüleri Oluşturmak İçin Önemli Komutlar

Marmara Kredi Döngüsünü gerçekleştirebilmek oldukça basittir. Sadece 4 komutu bilmeniz yeterlidir. Ancak kredi döngüsü var edebilmek veya bir sonraki düğüme aktarabilmek için yeterli aktif/kilitli koininiz olması gerekir. Bunun için kilitleme komutlarını kullanabilirsiniz.

Kredi döngüsü başlatma ve aktarma işlemlerini gerçekleştirmek için eşler (p2p) karşılıklı birbirinin public key adresini bilmek zorundadırlar.

- *marmarareceive*

Bu komut, bir keşideci veya cirantadan kredi almak için kullanılır. Bir keşideciden, yani ilk düğümden kredi istenirken, bunun benzersiz bir kullanımı vardır.

Senaryo 1: İki düğüm ilk kez bir kredi döngüsü oluşturuyor. Bu kredi döngüsü, piyasada bir mal veya hizmetin satışı için oluşturulabilir. Bu durumda, hamil (ürün/hizmeti satan kişi), keşideciden (ürün/hizmeti alan ve ödeme yapan kişi) aşağıdaki komutu yazarak kredi talebinde bulunmalıdır:
Linux:

```
./komodo-cli -ac_name=MCL marmarareceive senderpk amount currency matures '{"avalcount":"n"}'
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmarareceive senderpk amount currency matures {\"avalcount\":\"n\"}
```

senderpk keşidecinin (ürünü/hizmeti alan kişinin) pubkey adresidir

amount ödeme tutarıdır. Bu tutarın keşidecinin aktif fonunda bulunması gerektiğini, eğer yoksa keşideci tarafından marmaralock komutu ile aktifleştirilmesi gerektiğini unutmayın.

currency MARMARA yani MCL'dir.

matures ilgili kredinin sona erdiği zamandır, saatte 60 blok çarpı günde 24 saat, yani günde 1440 blok.

'{"avalcount":"n"}' avalistlerin sayısıdır, yani protokol 1 için '{"avalcount":"0"}'. Protokol 1 için n'yi 0 ile değiştirin.

Bu komut marmarareceive çağrısı bir onaltılık kod üretir. Bu HEXCODE'un hamil tarafından sendrawtransaction komutunu çalıştırarak doğrulanması (blokzincire kaydedilmesi) gerekir:
Linux:

```
./komodo-cli -ac_name=MCL sendrawtransaction HEXCODE
```

Windows:

```
komodo-cli.exe -ac_name=MCL sendrawtransaction HEXCODE
```

Bu komut yürütüldüğünde txid adında bir işlem kimliği oluşturulur. Kredi döngüsünü tamamlamak için bu txid'nin receiverpk (alıcı pubkeyi) ile birlikte keşideciye iletilmesi gerekir. Ancak bu iletişime bir alternatif, keşidecinin kendisine yapılan talepleri görmek için marmarareceivelist yöntemini kullanması olabilir. (Connector ile "Kredi Döngüleri – Gelen Döngü İstekleri" sekmesi altında yapılacak aramalarda seçilecek tarihlere gelen istekler görülebilir)

marmarareceivelist yöntemi keşideci tarafından aşağıdaki komutla yürütülür:
Linux:

```
./komodo-cli -ac_name=MCL marmarareceivelist pubkey maxage
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmarareceivelist pubkey maxage
```

pubkey Marmara Zincirine bağlı keşidecinin pubkey adresidir ve varsayılan maksimum değer 24x60'tır. Bu komutun yanıtı, belirli bir maksimum süre için ilgili pubkey'ler tarafından oluşturulan txid çiftlerinin bir listesidir.

- *marmaraissue*

Bu komut yalnızca krediyi oluşturan/veren ilk düğüm olan keşideci tarafından kullanılır. Bu sayede birinci hamile yani ikinci düğüme de bir kredi aktarılmış olur. Kredi döngüsüne ilişkin parametrelerin birçoğuna, keşideci ile ilk kredi hamili arasında karar verilir. marmaraissue yöntemi aşağıdaki argümanları alır:
Linux:

```
./komodo-cli -ac_name=MCL marmaraissue receiverpk '{"avalcount":"n", "autosettlement":"true"|"false", "autoinsurance":"true"|"false", "disputeexpires":"offset", "EscrowOn":"true"|"false", "BlockageAmount":"amount" }' requesttxid
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaraissue receiverpk "{\"avalcount\":\"n\", \"autosettlement\":\"true\"|\"false\", \"autoinsurance\":\"true\"|\"false\", \"disputeexpires\":\"offset\", \"EscrowOn\":\"true\"|\"false\", \"BlockageAmount\":\"amount\"}" requesttxid
```

receiverpk krediyi alacak olan hamilin pubkey'idir.

"avalcount":"n" protokol 1 için avalistlerin sayısıdır, örneğin {"avalcount":"0"}'. Marmara Kredi Döngüleri Protokol 1 kapsamında bu parametre için sıfır değeri verilmelidir.

"autosettlement":"true"|"false" Protokol 1'deki %100 teminatlandırma nedeniyle “true” olarak tanımlıdır.

"autoinsurance":"true"|"false" Protokol 1'deki %100 teminatlandırma nedeniyle “true” olarak tanımlıdır.

"disputeexpires":"offset" Protokol 1'deki %100 teminatlandırma nedeniyle 0 olarak ayarlanmıştır.

"EscrowOn":"true"|"false" EscrowOn, Protokol 1'deki %100 teminatlandırma nedeniyle false olarak ayarlanmıştır.

"BlockageAmount":"amount" } Protokol 1'de %100 teminatlandırma nedeniyle blokaj tutarı 0 olarak ayarlanmıştır.

requesttxid hamil tarafından oluşturulan ve keşideciye iletilen txid'dir.

Keşideci tarafından kredi döngüsünün tamamlanmasına yönelik tipik bir marmaraissue komutunun örneği aşağıda verilmiştir.
Linux:

```
./komodo-cli -ac_name=MCL marmaraissue receiverpk '{"avalcount":"0", "autosettlement":"true", "autoinsurance":"true", "disputeexpires":"0", "EscrowOn":"false", "BlockageAmount":"0" }' requesttxid
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaraissue receiverpk "{\"avalcount\":\"0\" \"autosettlement\":\"true\" \"autoinsurance\":\"true\" \"disputeexpires\":\"offset\" \"EscrowOn\":\"false\" \"BlockageAmount\":\"0\"}" requesttxid
```

Bu marmaraissue komutu bir onaltılık kod yanıtı döndürür ve şimdi keşidecinin, işlemin blockchain üzerinde yürütülmesini sağlamak için sendrawtransaction yöntemini aşağıdaki gibi yürütmesi gerekir:
Linux:

```
./komodo-cli -ac_name=MCL sendrawtransaction HEXCODE
```

Windows:

```
komodo-cli.exe -ac_name=MCL sendrawtransaction HEXCODE
```

Bu, keşideci ile hamil arasında bir kredi döngüsü oluşturur. Bir döngüde kilitlenen krediler, alışveriş sırasında bir şeyler satın almak için kullanılabilir. Keşideci ve hamil, kredi döngüsünün vadesi gelene kadar MCL fonlarına 3 kat stake yapma şansına sahip olur.

- *marmaracreditloop*

Hamil ile keşideci arasındaki kredi döngüsünü görüntülemek için aşağıdaki marmaracreditloop komutu çalıştırılabilir:
Linux:

```
./komodo-cli -ac_name=MCL marmaracreditloop txid
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaracreditloop txid
```

txid Marmara Kredi Döngüsü'nün baton transfer kimliğidir

- *marmaratransfer*

Senaryo 2: Önceki senaryodaki hamil, daha önce oluşturulan aynı kredi döngüsünden mal/hizmet satın alarak döngüde kilitlenen koinleri kullanmak istiyor. Böyle bir durumda, kredi sahibi bir döngü içerisinde bir krediyi devrettiğinde, o düğüm anında ciranta haline gelir. Ve bu şekilde, bir kredi döngüsündeki son düğüm sahibine daima hamil (hamil) denir. Diğer bir deyişle bütün cirantalar önceki durumda hamil rolünü alırlar. Burada dikkate alınması gereken nokta cirantaların krediyi hamile transfer ettikleri noktada 3 kat staking gücünü kaybetmeleri olacaktır.

Ciranta krediyi bir sonraki düğüme aktarmak için ```marmarareceive``` komutunu kullanır. Ciranta bu durumda artık hamil değildir:

Linux:

```
./komodo-cli -ac_name=MCL marmarareceive senderpk batontxid '{"avalcount":"n"}'
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmarareceive senderpk batontxid {\"avalcount\":\"n\"}
```

senderpk cirantanın (ürünü/hizmeti satın alan kişinin) pubkey adresidir

batontxid önceden oluşturulan kredi döngüsünün baton işlem kimliğidir

{"avalcount":"n"} avalistlerin sayısıdır, protokol 1 için, örneğin {"avalcount":"0"}

Bu marmarareceive çağrısı bir onaltılık kod üretir. Bu HEXCODE'un yeni sahibi tarafından sendrawtransaction komutunu çalıştırarak doğrulanması gerekir:
Linux:

```
./komodo-cli -ac_name=MCL sendrawtransaction HEXCODE
```

Windows:

```
komodo-cli.exe -ac_name=MCL sendrawtransaction HEXCODE
```

Bu komut yürütüldüğünde txid adında bir işlem kimliği oluşturulur. Kredi döngüsünü tamamlamak için bu txid'nin receiverpk ile birlikte cirantaya iletilmesi gerekir. Ancak bu iletişime bir alternatif, cirantanın kendisine yapılan istekleri görmek için kullanabileceği marmarareceivelist yönteminin kullanılması olabilir. Daha sonra ciranta, kredilerin yeni sahibine aktarılmasını sağlamak için aşağıdaki marmaratransfer komutunu çalıştırabilir:
Linux:

```
./komodo-cli -ac_name=MCL marmaratransfer receiverpk '{"avalcount":"n"}' requesttxid
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaratransfer receiverpk {\"avalcount\":\"n\"} requesttxid
```

receiverpk yeni hamilin pubkey adresidir.

"avalcount":"n" avalistlerin sayısını belirtir, Protokol 1 için '{"avalcount":"0"}' şeklinde verilmelidir.

requesttxid yeni hamil tarafından oluşturulan baton işlem numarasıdır.

Daha sonra ciranta, marmaratransfer komutundan kaynaklanan hex koduyla sendrawtransaction komutunu çalıştırır.

sendrawtransaction komutunun, blockchain üzerinde yürütülecek komutların sonuçlarını oluşturmak için marmarareceive, marmaraissue, marmaratransfer ve marmaralock'tan sonra kullanıldığını lütfen unutmayın. Kullanım senaryo 1 ve 2 boyunca sunulmuştur.

Bu sayede kredi döngüleri vade süresi içerisinde mal/hizmet satın almak için 1000'inci düğüme kadar dolaşabilmektedir.

- *marmaraholderloops*

Bu komut, ilk ve son yükseklik ile minimum ve maksimum miktarı ayarlayarak sınırlandırılabilen açık ve kapalı döngüleri döndürür. Tüm parametreler sıfıra ayarlanırsa yöntem mevcut tüm verileri döndürür. marmaraholderloops komutu aşağıda belirtilmiştir:

Linux:

```
./komodo-cli -ac_name=MCL marmaraholderloops firstheight lastheight minamount maxamount pubkey
```

Windows:

```
komodo-cli.exe -ac_name=MCL marmaraholderloops firstheight lastheight minamount maxamount pubkey
```

****

Kaynak: https://github.com/marmarachain/marmara/wiki/How-to-make-Marmara-Credit-Loops

Referanslar:

Marmara Kredi Döngülerinin nasıl çalıştığına dair daha detaylı bilgi için lütfen buradaki detaylı makaleye bakın: https://medium.com/@drcetiner/how-marmara-credit-loops-mcl-work-31d1896190a5

Önceden Derlenmiş İkili Dosyalardan Marmara Kurulumu

Önceden derlenmiş ikili dosyalarımızı indirebilir ve açabilirsiniz. Bu en basit yöntemdir ve dolayısıyla herhangi bir derleme prosedürü gerektirmez.

Linux

Sisteminizin güncel olduğunu doğrulayın.

```
sudo apt-get update
sudo apt-get upgrade -y
```

Bağımlılık paketlerini yükleyin

```
sudo apt-get install libgomp1
```

Sürümlerden Uygulamaları İndirme

Releases'e (sürümlere) gidin ve Assets'den (uygulamalardan - varlıklardan) Marmara linux ikili .zip dosyalarını indirin ve aşağıda verilen talimatları izleyin.

İndirdiğiniz dizinde veya dosyaları taşıdığınız - kopyaladığınız başka bir dizinde aşağıdaki talimatları izlediğinizden emin olun.

```
sudo apt install unzip
unzip MCL-linux.zip
sudo chmod +x komodod komodo-cli fetch-params.sh
./fetch-params.sh
```

Marmara Zincirini Başlatın ve Çalıştırın

Zinciri başlatmak için komut istemini açın ve aşağıdaki komut gruplarını yürütün:

```
./komodod -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -
addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -
ac_marmara=1 -ac_staked=75 -ac_reward=3000000000
```

Zinciri durdurmak:

```
./komodo-cli -ac_name=MCL stop
```

Cüzdanınızı Yedeklemek

wallet.dat dosyasının yedeklenmesi, bir kişinin varlıklarını barındırdığı için çok önemlidir. Bir Linux makinesinde dosya şurada bulunabilir: ~/.komodo/MCL/wallet.dat

Bu dosyayı yedeklemenin bir yöntemi, dosyanın bir kopyasını arşivlemektir.

```
# wallet.dat dosyasını kopyalayın
cp -av ~/.komodo/MCL/wallet.dat ~/wallet.dat

# wallet.dat dosyasının adını değiştirin
mv ~/wallet.dat ~/2020-08-09-wallet_backup.dat

# wallet.dat dosyasının bir arşiv dosyası oluşturun
tar -czvf ~/2020-08-09-wallet_backup.dat.tgz ~/2020-08-09-wallet_backup.dat
```

```
# Son olarak oluşturduğunuz bu dosyayı güvenli bir yerde saklayın
```

UFW’yi Etkinleştirmek

Bu adım MCL'yi yüklemek için gerekli değildir ancak sunucu güvenliği amacıyla kullanılabilir.

- UFW'nin durumunu şu şekilde kontrol edin:

```
sudo ufw status
```

- UFW paketi aşağıdaki komut çalıştırılarak kurulabilir

```
sudo apt install ufw
```

- UFW'yi etkinleştirin ve aşağıdaki komutları yürüterek bağlantılara izin verin

```
echo y | sudo ufw enable
sudo ufw allow ssh
sudo ufw allow "OpenSSH"
sudo ufw allow 33824
sudo ufw allow out 33824
```

MacOS

Komut satırı araçlarını yüklemek için bir terminalde aşağıdaki komutu verin:

```
xcode-select –-install
```

Releases (Sürümler) sayfasından Varlıklar (Assets) altındaki .zip dosyasını macOS makinenize indirin ve dosyaları çıkarın. Dizini terminaldeki komodod & komodo-cli konumuna değiştirin.

Aşağıdaki komutu çalıştırın:

```
./fetch-params.sh
```

Marmara Zincirini Başlatın ve Çalıştırın

Zinciri başlatmak için komut istemini açın ve aşağıdaki komut gruplarını yürütün:

```
./komodod -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -
addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -
ac_marmara=1 -ac_staked=75 -ac_reward=3000000000
```

Zinciri durdurmak için:

```
./komodo-cli -ac_name=MCL stop
```

Cüzdanınızı Yedekleyin

Wallet.dat dosyanızı yedeklemek çok önemlidir. Bu dosyayı yedeklemenin bir yöntemi, dosyanın bir kopyasını arşivlemektir.

MacOS'ta Wallet.dat dosyası burada bulunur: ~/Library/Application\ Support/Komodo/wallet.dat

Windows

Sürümlerden Dosya İndirme

.zip dosyasını Sürümler sayfasından buradaki Varlıklar (Assets) bölümünden Windows bilgisayarınıza indirin ve dosyaları (komodod.exe & komodo-cli.exe) masaüstünde MCL adı verilen yeni bir klasöre veya tercih ettiğiniz başka bir konuma çıkartın ve yerleştirin (konumu unutmayın ve bunu kullanın).
Bu kılavuz için Masaüstünde MCL dizinini kullanıyoruz.

Zcash Parametreleri için Dizin Oluşturun

Komut satırı istemini açın ve aşağıdaki komutu çalıştırın:

```
mkdir “%HOMEPATH%\AppData\Roaming\ZcashParams”
```

Aşağıdaki dosyaları indirin ve bunları yeni oluşturulan dizine taşıyın, örneğin MCL.

- sprout-proving.key
- sprout-verifying.key
- sapling-spend.params
- sapling-output.params
- sprout-groth16.params

Marmara Zincirini Başlatın ve Çalıştırın

Zinciri başlatmak için komut istemini açın ve aşağıdaki komut gruplarını yürütün:

```
cd Desktop\MCL\

komodod.exe -ac_name=MCL -ac_supply=2000000 -ac_cc=2 -addnode=5.189.149.242 -addnode=161.97.146.150 -addnode=149.202.158.145 -addressindex=1 -spentindex=1 -ac_marmara=1 -ac_staked=75 -ac_reward=3000000000
```

Zinciri durdurmak için:

```
cd Desktop\MCL\
komodo-cli.exe -ac_name=MCL stop
```

Cüzdanı Yedeklemek

wallet.dat dosyasını yedeklemek çok önemlidir. Bu dosyayı yedeklemenin bir yöntemi, bir kopyasını oluşturup arşivlemektir.

Windows makinesinde dosya aşağıda bulunur:

%HOMEPATH%\AppData\Roaming\Komodo\MCL\wallet.dat

Marmara Kredi Döngüleri Oluşturmak

Buradaki kullanım kılavuzunda anlatıldığı gibi pubkey ile zinciri başlatın veya https://github.com/marmarachain/marmara/wiki/Getting-Started-with-Marmara

****

Kaynak: https://github.com/marmarachain/marmara/wiki/Install-Marmara-from-Precompiled-Binaries

Marmara Blokzincirini Anlamak

Marmara Zinciri, %25 Proof of Work (PoW) ve %75 Proof of Stake (PoS) kullanan hibrit mutabakat mekanizması ile blokzinciri dünyasında öne çıkıyor. Bu yenilikçi karışım, verimli ve ölçeklenebilir bir ağı desteklerken güvenli, merkezi olmayan kontrol sağlar. Blok zincirlerinin çoğu Proof of Work (PoW) veya Proof of Stake (PoS) mutabakat mekanizmalarını kullanır. PoW, blok zincirine yeni bir blok eklemek için karmaşık bulmacaları çözmeyi içerir (bunu bir dijital madencilik süreci gibi düşünün). Öte yandan PoS, yeni bir bloğu doğrulamak için belirli miktarda koinin tutulmasını ve kilitlenmesini içerir (noterlerin ortak bir fondaki paylarına göre ama kendi kendini finanse eden bir mekanizma ile ödül aldıklarını hayal edin).

Marmara Kredi Döngüleri (MCL) Çözümü

Marmara Zinciri'nin kalbinde, geleneksel finansal araçlara vadelerinden önce bile benzeri görülmemiş likidite ve fayda sağlayan bir sistem olan MCL yer almaktadır. Ama hepsi bu kadar değil. MCL, staking için 'etkinleştirme' ve 'kilitleme' konusundaki benzersiz yaklaşımı benimsiyor ve farklı derecelerde stake gücü sağlıyor - etkinleştirilenler için 1x ve kredi döngülerinde kilitli olanlar için 3x. En iyi kısım? Etkinleştirilmiş koinlerin kilidi her zaman açılabilir ve geleneksel staking sistemlerinde duyulmamış bir likidite seviyesi sunarken, kredi döngülerinde kilitlenen koinler, vadeli çeklerle benzer şekilde cirolanarak gerçek ekonomide dolaşıma girmeye devam edebilir.

Kredi Döngüleriyle Devrim Yaratan İşlemler

Kredi döngüleri, vadeli veya senetlerin dijital versiyonları gibi davranarak benzersiz avantajlar sunar. Kredi döngülerinde kilitlenen koinler yalnızca 3 kat staking gücüne sahip olmakla kalmaz, aynı zamanda cirolanarak gerçek ekonomide dolaşıma girebilir. Bu şekilde, gerçek dünyadaki işlemlere hizmet ederken keşideci ve hamilin staking gücüne katkıda bulunmaya devam ederler. Vade tarihinde, otomatik olarak açılırlar.

Madencilik ve MCL Kazanmak

Marmara Zinciri'nin yerel para birimi olan MCL koinleri birden fazla yoldan kazanılabilir. %25 PoW mutabakat mekanizması sayesinde madencilik yapılabilir. Alternatif olarak Marmara Kredi Döngüleri sistemi içerisinde mal ve hizmet alım satımı yapılarak da kazanılabilir. Bu, MCL koinlerinin faydasını ve çekiciliğini daha da artırır.

****

Kaynak:
https://marmara.io/tr/blog/marmara-kredi-donguleri-1/marmara-zinciri-ve-marmara-kredi-donguleri-icin-nihai-basitlestirilmis-klavuz-54

Marmara Blockchain Staking'in Gücünü Açığa Çıkarın

Maksimum APR için Yatırım Stratejileri

Marmara Blockchain'in heyecan verici dünyasına hoş geldiniz! Size Marmara Blockchain staking mekanizması ve en yüksek Yıllık Yüzde Oranı (APR) elde etmenize yardımcı olacak yatırım stratejileri hakkında önemli bilgiler sunmaktan büyük mutluluk duyuyoruz. Bu kapsamlı rehberde, Marmara'nın yenilikçi staking sistemini inceleyecek ve dikkate değer getiriler elde etmenize yardımcı olacak değerli içgörüler sunacağız.

Marmara Blockchain Staking Mekanizması

Marmara Blockchain, %25 madencilik (İşlem Kanıtı) ve %75 staking (Paylaşımlı Kanıt) olmak üzere karmaşık bir staking sistemine sahiptir. Staking kısmı, Aktif Staking ve Kredi Döngüsü Staking olmak üzere iki bölüme ayrılmıştır.

1. **Aktif Staking:** Coin'lerinizi aktif staking fonlarında kilitleyerek, %75 Paylaşımlı Kanıt arasında 1x staking gücü elde edersiniz. Aktif staking, toplam staking ödüllerinin %25'ini oluşturur. Coin'ler, istediğiniz zaman kilidini açarak, yatırımlarınızı yönetmede daha fazla esneklik sunar.
2. **Kredi Döngüsü Staking:** Kredi Döngüsü Staking: Coin'lerinizi kredi döngülerinde kilitleyerek, 3x staking gücü elde edersiniz. Bu, toplam staking ödüllerinin %75'inin kredi döngülerinden geldiği anlamına gelir. Bu kategoride üretilen staking ödülleri, aktif staking'den önemli ölçüde daha yüksektir.
3. **Optimal APR İçin Yatırım Stratejileri:** APR'nizi en üst düzeye çıkarmak için, dört kritik parametreyi göz önünde bulundurmanız gerekir: aktif staking'de kilitlenmiş toplam coin'ler, tüm zincirdeki kredi döngülerinde kilitlenmiş toplam coin'ler ve aktif ve kredi döngülerinde kilitlenmiş coin'lerinizin sayısı. İşte Marmara Blockchain yatırımlarınızdan en iyi şekilde yararlanmanıza yardımcı olacak üç yatırım stratejisi:
    - **Blokzinciri Parametrelerini İnceleyin:** Blockchain Parametrelerini İnceleyin: Marmara Blockchain'deki hem aktif staking'de hem de kredi döngülerinde kilitlenmiş toplam coin'lere dikkat edin. Bu bilgilere Marmara Zincir Keşif sitesinden erişilebilir. Bu parametreleri düzenli olarak izleyerek, coin'lerinizi hangi kategoride ve ne zaman kilitleyeceğinize dair bilinçli kararlar verebilirsiniz.
    - **Yatırımınızı Çeşitlendirin:** APR'nizi optimize etmek için etkili bir strateji, yatırımlarınızı hem aktif staking hem de kredi döngüleri arasında çeşitlendirmektir. Coin'lerinizin bir kısmını, kilidini açma özelliği nedeniyle daha fazla esneklik sunan aktif staking'e ayırın ve geri kalan coin'leri daha yüksek staking gücünden yararlanmak için kredi döngülerine yatırın.
    - **Potansiyel Ödüllerinizi Hesaplayın:** Marmara Staking Calculator kullanarak, farklı yatırım senaryolarına göre günlük ve yıllık kazançlarınızı ve yıllık ROI'nizi tahmin edin. Aktif ve kredi döngülerinde kilitleyeceğiniz coin'lerin çeşitli kombinasyonlarıyla deneyerek, en yüksek APR'yi sağlayan en uygun dağıtımı belirleyebilirsiniz.

Sonuç

Marmara Blockchain, stratejik yatırım kararlarıyla geri dönüşlerinizi en üst düzeye çıkarmanıza olanak tanıyan benzersiz ve kârlı bir staking mekanizması sunar. Staking sistemini anlayarak, yatırımlarınızı çeşitlendirerek ve Marmara Staking Calculator, kullanarak, Marmara Blockchain'in tam potansiyelini açığa çıkarabilir ve yatırımlarınızdan etkileyici bir APR elde edebilirsiniz. İyi stakingler!

****

Kaynak:
https://marmara.io/tr/blog/marmara-kredi-donguleri-1/marmara-blockchain-staking-in-gucunu-acga-ckarn-27

Marmara Chain'de MCL Staking Gücünü Keşfedin: Kripto Ekonomisinde Benzersiz Avantajların Kilidini Açın

Komodo Platform'unun bir çatalı olan Marmara Chain, benzersiz ve güçlü staking mekanizmasıyla staking dünyasını devrim niteliğinde değiştiriyor. Esneklik sunan yenilikçi MCL staking sistemi, geleneksel staking platformlarına göre sayısız avantaj sunmaktadır. Bu yazıda, Marmara Chain'de MCL staking'in inanılmaz faydalarını ve kripto para dünyasındaki dönüşümü ele alacağız.

1. Minimum Staking Gereksinimi Yok: Herhangi Bir Miktarı Staking Yapma Özgürlüğü

Ethereum gibi diğer staking platformları, staking'e katılmak için en az 32 ETH gerektirirken, Marmara Chain'in minimum staking gereksinimi yoktur. Bu, kullanıcıların herhangi bir MCL miktarıyla staking yapabileceği ve staking ödülleri kazanmak isteyen herkes için erişilebilir ve uygun maliyetli olduğu anlamına gelir.

2. Esnek Staking: Coin'lerinizi İstediğiniz Zaman Kilidini Açın

Marmara Chain'in staking sistemi, kullanıcıların ihtiyaç duyduklarında stake edilen koinlerin kilidini açmalarına olanak tanır. Bu esneklik, staking süresinin sonuna kadar coin'lerin kilidini açma seçeneği olmayan sabit süreli diğer staking platformlarıyla karşılaştırıldığında önemli bir avantajdır. Marmara Chain'in staking sistemi, kullanıcıların varlıkları üzerinde kontrollerini korumalarını ve ihtiyaç duyduklarında kullanmalarını sağlar.

3. *Marmara Kredi Döngüleri: Staking Gücünüzü 3x Artırın*

Marmara Chain'in eşsiz staking mekanizması, coin'leri staking için kilitleme konusunda iki yol sunar: etkinleştirilmiş (aktif) staking modu ve Kredi Döngüleri. Etkinleştirilmiş staking modu, 1x staking gücü sağlarken, Kredi Döngüleri kullanıcılara etkileyici 3x staking gücü sunar. Bu artan staking gücü, ödülleri önemli ölçüde artırabilir ve MCL staking'i katılımcılar için daha karlı hale getirir.

4. Gerçek Ekonomide Staking Coin'lerin Dolaşımı

Marmara Chain'in staking sisteminin öne çıkan özelliklerinden biri, staking coin'lerin gerçek ekonomide dolaşımına olanak tanımasıdır. Kredi Döngüleri'nde kilitli coin'ler, vadeleri gelmeden önce vadeli çekler veya senetler gibi basit bir cirolama yöntemiyle başkasına devredilebilir. Bu, MCL koinlerin ve Marmara Chain'in gerçek ekonomide kullanılabilir olmasını sağlar ve diğer tüm kripto para zincirlerinden benzersiz kılar. Koinleri Kredi Döngüsüne devrederek, kullanıcı staking haklarını kaybeder ve bu nedenle koinlerin sadece gerçek ekonomide kullanılması anlamlı hale gelir.

5. *%100 Teminatlandırma: Güvenli ve Güvenilir Bir Sistem*

Marmara Kredi Döngüleri, Marmara blok zincirinde tamamen teminatlandırılmıştır ve güvenli ve güvenilir bir staking sistemi sağlar. Bu güvenlik seviyesi, vadeli çekler gibi geleneksel kredi araçlarıyla sıklıkla ilişkilendirilen karşılıksızlık ve geri ödeme riskini ortadan kaldırır ve MCL stake edenler için huzur verir.

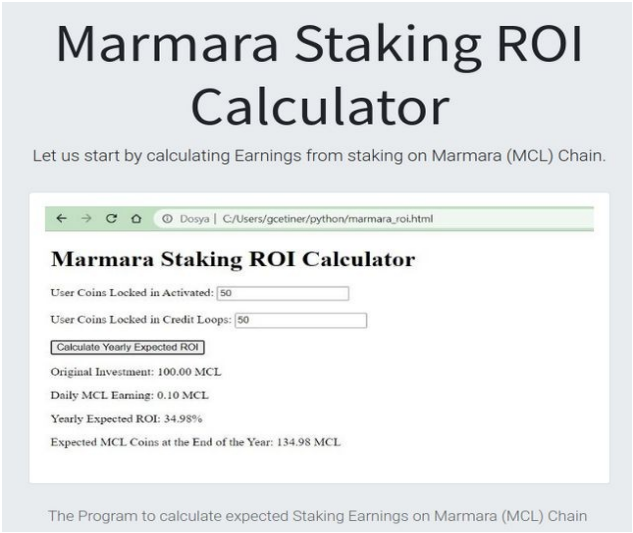
Sonuç:
Marmara Chain, saniyeler içinde blok süreleri ve daha fazla blok tutma gereksinimi nedeniyle güçlü bilgisayarlar gerektiren diğer staking platformlarından farklıdır. Marmara Chain'in staking sistemi daha hafif donanım gereksinimleriyle çalışır ve bu da staking'e katılmak isteyen kullanıcılar için daha erişilebilir ve maliyet-etkin hale getirir. Sonuç olarak, Marmara Chain'in staking sistemi, kullanıcıların herhangi bir miktarı stake etme, istedikleri zaman coin'lerin kilidini açma, 3x staking gücü sunan Kredi Döngüleri ve gerçek ekonomide dolaşan coin'ler gibi eşsiz avantajlar sunarak kripto para dünyasında öne çıkmaktadır. Ayrıca, daha hafif donanım gereksinimleriyle daha geniş bir kitleye hitap etmektedir. MCL staking'i ve Marmara Chain'i keşfederek, kripto ekonomisine yeni bir soluk getiren bu benzersiz fırsatlardan yararlanabilirsiniz.

****
Kaynak:
https://marmara.io/tr/blog/marmara-kredi-donguleri-1/marmara-chain-de-mcl-staking-gucunu-kesfedin-21

Marmara (MCL) Koin Neden Öne Çıkıyor?

Kripto paraların hızla gelişen dünyasında, staking gibi pasif gelir fırsatları yatırımcılar arasında giderek daha popüler hale gelmektedir. Sayısız blockchain platformunun çeşitli staking ödülleri sunmasıyla, birikimlerinizi değerlendirebileceğiniz doğru yatırımı bulmak zor olabilir. İşte burada, etkileyici %34,98 APR* staking ödülleriyle Marmara (MCL) Coin, rakiplerinden gerçekten sıyrılıyor.

Bu yazıda, Marmara Zinciri'nin diğer önde gelen blockchain platformlarına kıyasla sunduğu staking ödüllerini karşılaştıracağız. Ayrıca, Marmara Zinciri'nin benzersiz özelliklerinin pasif gelirlerini en üst düzeye çıkarmak isteyen yatırımcılar için neden ideal bir seçenek olduğunu da tartışacağız.



Marmara Zinciri'nin şu anki etkileyici %34,98 APR'si, staking için mükemmel bir seçenek olmasının tek nedeni değil. İşte üstünlüğüne katkıda bulunan bazı temel faktörler:

1. **Yenilikçi Kredi Döngüleri Sistemi:** Marmara Zinciri'nin benzersiz Kredi Döngüleri sistemi, kullanıcıların kredi döngülerine katılarak daha yüksek ödüller kazanmalarını sağlar ve size 3 kat staking gücü sunar.
2. **Merkeziyetsizlik ve Güvenlik:** Marmara Blokzinciri, sağlam ve güvenli bir ağa sahip olarak, kendi ürettiğiniz MCL koinlerinizi olası tehditlere karşı korur.
3. **Kullanıcı Dostu Staking Süreci:** Kullanıcı Dostu Staking Süreci: MCL koinleri staking yapmak kolay ve anlaşılır bir süreçtir, hem acemi hem de deneyimli yatırımcılar için erişilebilir.
4. **Staking Devrimi:** Marmara Zinciri'nin staking mekanizması, %75 Proof of Stake (PoS) içerir; bu da çoğu bloğun staking yoluyla üretildiği anlamına gelir. Diğer staking blockchain'lerine kıyasla, Marmara Blokzinciri'nin sunduğu avantajlar, kripto meraklıları için tercih edilen seçenek haline gelmesini sağlar.
5. **Minimum Staking Gereksinimi Yok:** Ethereum gibi, staking için minimum 32 ETH gerektiren platformların aksine, Marmara Zinciri kullanıcıların herhangi bir miktarda MCL tokeni staking yapmalarına olanak tanır. Bu kapsayıcı yaklaşım, daha fazla kişinin finansal engellerle karşılaşmadan staking sürecine katılmalarını sağlar.

6.**Esnek Kilitleme Açma:** Marmara Zinciri, stake edilen koinlerin kilidini açma konusunda eşsiz bir esneklik sunar. Kullanıcıların önceden belirlenmiş bir süre beklemesi gereken diğer staking platformlarının aksine, Marmara Zinciri kullanıcıların stake edilen koinlerinin kilidini istedikleri zaman açmalarına izin verir ve böylece likiditeyi artırır.

7.**Marmara Kredi Döngüleri – 3x Staking Gücü:** Marmara Zinciri'nin yenilikçi Kredi Döngüleri, geleneksel aktif staking'e kıyasla inanılmaz bir 3x staking gücü sağlar. Bu artırılmış staking gücü, platformun benzersiz Kredi Döngüleri sistemine katılan kullanıcılar için faydalıdır.

8.**Gerçek Ekonomide Kullanılabilir Kredi Döngüleri:** Marmara Kredi Döngüleri, vadeli çekler ve senetlerden esinlenerek, staking sırasında koinlerin gerçek ekonomiler içinde dolaşımını sağlar. Kullanıcılar, Kredi Döngüleri sözleşmelerinde kilitlenmiş koinleri vade tarihlerinden önce cirolayabilir veya dolaşıma sokabilir, böylece MCL, gerçek dünya işlemleri için pratik bir çözüm haline gelir.

9.**Güçlü Donanıma Gerek Yok:** Marmara Blokzinciri'nin staking mekanizması, yüksek performanslı bilgisayar donanımına ihtiyaç duymaz, çünkü bir dakikalık blok süresi verimli blok üretimini sağlar. Bu erişilebilirlik, pahalı ekipmanlara yatırım yapmaktan endişe etmeden stakinge katılmak isteyen daha fazla kişiye olanak tanır.

10.**Marmara Blokzinciri ile Staking Geleceği:** Marmara Zinciri'nin benzersiz staking mekanizması ve Marmara Kredi Döngüleri ile birleşerek, kripto meraklıları için eşsiz bir deneyim sunar. Artırılmış staking gücü, esnek kilitleme açma ve gerçek dünya uygulanabilirliği sağlayarak, Marmara Zinciri kripto alanında bir oyun değiştirici olmaya hazırdır.

11.**Aktif Geliştirme ve Destek:** Marmara Zinciri'nin kendine özgü geliştirici ekibi ve destekleyici topluluğu bulunur, bu da platformun zamanla büyümeye ve gelişmeye devam etmesini sağlar.

12.**Gelecekteki Büyüme Potansiyeli:** Yenilikçi özellikleri ve sağlam temelleriyle Marmara Blokzinciri, blockchain teknolojisinin rekabetçi dünyasında sürekli büyüme ve başarı için iyi bir konumdadır.

Sonuç

Sonuç olarak, Marmara (MCL) Koin, diğer önde gelen blockchain platformlarına kıyasla benzersiz staking ödülleri sunar. Şu anki etkileyici %34,98 APR ile birlikte, yenilikçi Kredi Döngüleri sistemi, sağlam güvenlik, kullanıcı dostu staking süreci ve gelecekteki büyüme potansiyeli, Marmara Zinciri'ni pasif gelirlerini en üst düzeye çıkarmak isteyen yatırımcılar için olağanüstü bir seçenek haline getirir.

Kripto para yatırımlarınızı Marmara (MCL) Coin ile büyütme fırsatını kaçırmayın. Bugün gelişen Marmara Zinciri topluluğuna katılın ve üstün staking ödüllerinin faydalarını kendiniz deneyimleyin!

****

Kaynak:

https://marmara.io/tr/blog/marmara-kredi-donguleri-1/marmara-mcl-coin-ile-ustun-staking-odullerini-kesfedin-kapsaml-bir-karslastrma-23

Marmara Connector

Marmara Connector Kurulumu

Kaynak Kodundan Marmara Connector Kurulumu

Yürütülebilir dosya için
Proje dosyasındaki terminali açın. Yani src'nin olduğu yer.

- Test Denemesi

```
fbs run
```

- Yürütülebilir dosya oluştur

```
fbs freeze
```

Marmara Connector Geliştirmeye Başlarken

Geliştirme Ortamının Hazırlanması

Önkoşullar

PYTHON3 -> Fbs paketleme için python3.6 gereklidir
PyQt5
QT Designer
Güncel sürüme sahip pip kütüphanesi

Linux

İşletim sisteminin güncel ve yükseltilmiş olduğundan emin olun:

```
sudo apt-get update
sudo apt-get upgrade
```

Bağımlılıklar oluşturmak için gereken araçları yükleyin:

```
sudo apt install git python3 python3-pip python3-venv
```

Depoyu yerel makinede istediğiniz bir konuma kopyalayın:

```
git clone https://github.com/marmarachain/marmara-connector.git
```

Sistem çapında bir PYQT5 ve Designer Kurulumu için terminalde aşağıdaki komutları verin:

```
pip3 install pyqt5==5.9.2
sudo apt-get install qttools5-dev-tools
sudo apt-get install python3-pyqt5
sudo apt-get install pyqt5-dev-tools
```

Projede sanal bir ortamla çalışmak için projenin çalışma dizini dışında aşağıdaki komutları verin:

```
python3 -m venv venv
source venv/bin/activate
pip install pip==21.2.4
pip install -r requirement
```

Projeyi çalıştırmak için /src/main/python/ dizini altında mainApp.py dosyasını çalıştırın:

```
python3 mainApp.py
```

Windows
Bağımlılıklar oluşturmak için gereken aşağıdaki araçları yükleyin:

```
git python3 python3-pip python3-venv
```

QT5 Designer Kurulumu:
Aşağıdaki komutu kullanarak "pyqt5-tools"un en son sürümünü yükleyin:

```
pip install pyqt5-tools --pre
```

"designer.exe" ...Lib\site-packages\qt5_applications\Qt\bin dizinine yüklenecek

Depoyu yerel makinede istediğiniz bir konuma kopyalayın:

```
git clone https://github.com/marmarachain/marmara-connector.git
```

Dizini marmara-connector olarak değiştirin. Geçerli dizinde sanal bir ortam oluşturun:

```
python3 -m venv venv
```

Sanal ortamı etkinleştirin:

```
call venv\scripts\activate.bat
```

Sanal ortam aktifken aşağıdaki adımlara devam edin.
Gerekli kütüphaneleri yükleyin:

```
pip install fbs PyQt5==5.9.2
pip install pip==21.0.1
pip install -r requirement
```

Kaynaklar
Windows işletim sistemi ortamını farklı python sürümleriyle ayarlamak için buraya göz atın:
https://medium.com/swlh/how-to-set-up-a-virtual-environment-with-a-different-python-version-on-windows-10-9900eb0acf9a

Sürümlerden Marmara Connector Kurun

Lütfen Marmara Connector yüklemek için Releases'den Assets'i bulun.

- **Ubuntu Kurulumu**

İlgili .zip dosyasını indirin ve dosyayı açın.
Uygulama dosyasının çalışmasını etkinleştirmek için:

```
chmod +x MarmaraConnector
```

- **Windows Kurulumu**

İlgili .exe dosyasını indirin. Uygulama yüklendikten sonra yönetici olarak çalıştırın.

Marmara Connector Yazılım Özellikleri

Kısaltmalar
OS: İşletim Sistemi
MBC: Marmara Blokzincir
UI: Kullanıcı Arayüzü

Tanımlar
Keşideci: Kredi döngüsünde ilk olarak kredi oluşturan kişi. Kredi döngüsünde ilk düğümü oluşturan kişidir. Kredi, vade tarihine kadar %100 veya sıfır teminatla teminatlandırılabilir.

Hamil (Sahip): Kredi döngüsündeki son düğüme her zaman hamil (sahip) denir. Birisi bir döngü içinde kredi aktardığında, bu düğüm anında ciranta haline gelir.

Ciranta (Cirolayan): Kredi döngüsünün ilk düğümü olan keşideci ile son düğüm arasına giren ve krediyi bir sonraki düğüme aktaran diğer tüm düğümler.

Vade: Bir kredinin sona erme süresi. Tahminde bloklar olarak ölçülür. Günlük blok 1440'tır (saatte 60 blok çarpı günde 24 saat). Bir kredinin 100 gün vadeli olduğunu varsayarsak, vadesi 1440×100 yani 144.000 bloktur.

Mahsuplaşma: Bir kredi vadesi geldiğinde, bir döngüdeki son düğüm olan hamiline ödeme yapılır. Mahsuplaşma otomatik veya manuel olabilir.

Eskrov: Protokol 2 için güvene dayalı taraflar. EscrowOn eğer false ise, %100 teminatlandırma kullanılır ve uzlaşma otomatik olur. Tam güvenli sürüm olarak çalışan Protokol 1'de eskrovlara (emanetçilere) gerek yoktur.

Avalist: Avalistler, MCL'li ciranta veya cirantacıları ek teminatlandırma olarak destekler ve bu destekle 3 kat staking kazanabilirler. Geri ödeme yapılmaması durumunda fonları kullanılır. Avalistler yalnızca Protokol 2'de mevcuttur. Avalcount parametresi Protokol 1 için her zaman sıfırdır.

Blokaj Tutarı: Bu 2. protokolün bir terimidir. Hamil cirantadan bir miktar teminat koymasını isteyebilir. Bu durumda, ciranta protokol 2'de 3 kat stake avantajından yararlanır.

Zaman Aşımı: Protokol 2'deki kredi döngülerinde olası bir karşılıksızlık problemini çözmede verilen süreye denir. Bu tarihe kadar mashuplaşmanın (settlement) yapılmış olması gerekir. Bu tarihten sonra hamil hak iddia edemez. Keşideci, protokol 2 kapsamında teminatlandırma olmadan veya yetersiz teminatlandırma ile kredi oluştururken bu süreyi blok sayısıyla belirleyebilir. Bu süre sona ermeden escrovun, itfa edilememe sorununu çözmek için keşideci ile yapılan anlaşmaya göre tüm işlemleri yapması gerekir. Aksi takdirde eskrov sistemde cezalandırılır.

İşlevsel Olmayan Sistem Gereksinimleri
- Bu yazılım, Windows ve Linux işletim sistemi için dağıtımları olan masaüstü tabanlı bir GUI (Grafik arayüz) uygulamasıdır.
- Yazılım, GUI tasarımı için PyQt kütüphanesinin yanı sıra Qt Designer kullanılarak Python'da geliştirilmiştir.
- Hatalar, yeni özellikler, iyileştirmeler, Marmara Connector github sayfasındaki github sorunları üzerinden ele alınmaktadır.

- Dağıtım paketleri, Sürümler (Releases) sayfasında ayrıntılarla birlikte verilmektedir.
- Wiki, kaynak koddan Connector oluşturmak ve geliştiricilere açık kaynak kod tabanına katkıda bulunma konusunda rehberlik etmek için oluşturulmuştur.
- Kod tabanına yapılan katkılar, çekme istekleri yoluyla memnuniyetle karşılanır.

Fonksiyonel Sistem Gereksinimleri

- Yazılım, Marmara Protokol 1'i temel almaktadır.
- Marmara Connector, yerel veya uzak bir makine/platform üzerinde çalışan Marmara blokzincirine bağlantı sağlar.
- Yerel platform Linux, Windows veya MacOS tabanlı işletim sistemidir.
- Uzak platform Linux tabanlı işletim sistemidir.
- Masaüstü uygulaması, cüzdan işlemlerinin yanı sıra Marmara Kredi döngüsü işlemlerini de gerçekleştirmek için kullanılmaktadır.

Aşağıdaki parametreler, Marmara Protokol 1'de kararlaştırıldığı gibi kod bazında aşağıdaki şekilde ele alınacaktır.

- **autosettlement:** Protokol 1'deki %100 teminatlandırma nedeniyle true olarak ayarlanmıştır.

- **autoinsurance:** Protokol 1'deki %100 teminatlandırma nedeniyle true olarak ayarlanmıştır.

- **disputeexpires:** Protokol 1'deki %100 teminatlandırma nedeniyle 0 olarak ayarlanmıştır.

- **EscrowOn:** Protokol 1'deki %100 teminatlandırma nedeniyle false olarak ayarlanmıştır.

- **BlockageAmount:** Protokol 1'deki %100 teminatlandırma nedeniyle 0 olarak ayarlanmıştır.

Genel Kısıtlamalar/Kurallar

- Bu belgede tanımlanan kullanıcılar, açıkça belirtilmediği sürece keşideci, hamil veya ciranta olabilirler.
- Vade bloğunu temsil eden blok yükseklikleri, tarih saat verilerine (veya tersi) dönüştürülmelidir.
- Datetime alanları için en erken tarih, MBC'nin başlangıcından önce olmamalıdır. En son datetime geçerli datetime’dan daha ileri olmamalıdır.
- Tarih-saat alanları için, tarihin kolay seçimi için kullanıcı arayüzünde bir takvim görünümü sunulmalıdır.
- Doğrudan MBC'den alınan mesajlar dışında kullanıcı arayüzündeki her kelimenin çevrilebilir olması gerekir.
- MBC'ye yapılan ve MBC'den alınan eylemler ve çağrılar, uygun olan bir yerde günlük dosyasına kaydedilmeli ve kullanıcı tarafından erişilebilir olmalıdır.

Sunucu Tanımı

- Kullanıcı, sunucu adı, sunucu kullanıcı adı, ip adresi ve sunucu şifresi alanlarına sahip bir sunucu kaydı oluşturabilmelidir.
- Kullanıcının kaydettiği sunucu adı, sunucu kullanıcı adı ve ip adresi alanları uygulama tarafından okunabilir, yazılabilir bir dosyada saklanmalıdır. Şifre saklanmamalıdır.
- Kullanıcı tüm sunucu detaylarını listeden görebilmeli (okuyabilmeli) ve

güncelleyebilmelidir.
- Kullanıcının listeden bir sunucuyu silebilmesi gerekir.

İş Kuralları/Kısıtlamalar/Varsayımlar (Sunucu Tanımı)

- Sunucu sayfasında; sunucu adı, sunucu kullanıcı adı, ip adresi ve sunucu şifresi zorunlu alanlar olmalı ve bunlardan herhangi biri boş bırakılamaz olmalıdır.
- Kullanıcının kayıtlı sunucu alanlarından herhangi birini güncellemek istemesi durumunda, kullanıcıyı ilgili güncellemeden emin olup olmadığı konusunda uyarmak için bir mesaj açılmalıdır.
- Kullanıcının bir sunucuyu silmek istemesi durumunda, silme işleminden emin olup olmadığı konusunda kullanıcıyı uyarmak için bir mesaj açılmalıdır.
- Sunucu detayları, kullanıcı GUI uygulamasını kapatsa bile bir dosyada saklanmalıdır.
- Sunucu ayrıntılarını içeren veri dizini, ilgili ana bilgisayarda uygulama kaldırılsa veya yükseltilse bile kalmalıdır.

Kişiler Tanımı

- Kullanıcı, adı, cüzdan adresi ve pubkey alanlarına sahip bir kişi oluşturabilmelidir.
- Kullanıcı tarafından kaydedilen isim, cüzdan adresi ve pubkey alanları uygulama tarafından okunabilen, yazılabilen bir dosyada saklanmalıdır.
- Kullanıcı, tüm kişilerini listeden görebilmeli (okuyabilmeli) ve güncelleyebilmelidir.
- Kullanıcı listeden bir kişiyi silebilmelidir.
- Kişiler, kullanıcı GUI uygulamasını kapatsa bile bir dosyada saklanmalıdır.
- Kullanıcı, mümkün olduğu durumlarda kredi döngüsü işlemleri yaparken, kişiler listesine eklediği kişinin adını görebilmelidir.

İş Kuralları/Kısıtlamalar/Varsayımlar (Kişiler Tanımı)

- İletişim sayfasında; isim, cüzdan adresi ve pubkey alanları zorunlu alanlardır ve boş bırakılamaz olmalıdırlar.
- Kullanıcının, kişilerin herhangi bir alanını güncellemesi durumunda, kullanıcıyı ilgili güncellemeden emin olup olmadığı konusunda uyarmak için bir mesaj açılmalıdır.
- Kullanıcının bir kişiyi silmek istemesi durumunda, silme işleminden emin olup olmadığı konusunda kullanıcıyı uyarmak için bir mesaj açılmalıdır.

MBC Kurulumu

- Programın MBC'nin kurulu olup olmadığını kontrol etmesi gerekiyor.
- Kullanıcı, kurulu değilse MBC'yi platforma (yerel veya uzak) yükleyebilmelidir.
- MBC'nin önceden kurulmuş olması durumunda, kullanıcıdan src klasörüne bir yol sağlaması istenebilir. *
- Yükleme sırasında, ilgili bilgileri görüntüleyen bilgi kutusuyla birlikte önyüklemeyi indirmek için bir önyükleme seçeneği sağlanmalıdır (Önyükleme (bootstrap) önerilmez).
- Kurulum sırasında, kullanıcının süreç ve ilerleme hakkında bilgilendirilmesi gerekir. (İndeksleme ilerlemesi vb.)

* Eğer Marmara Chain kurulumunu önceden yaptı iseniz, Marmara Connector linux işletim sistemlerinde home klasörünüzdeki marmara dizini altındaki src dizininde comodod ve komodo-cli dosyalarını arayacaktır. (veya home dizininizde marmara/src & komodo/src klasörlerine göz atacaktır) Kurulumu yaptıktan sonra "Local" seçeneğine tıkladığınızda daha önce kurmuş olduğunuz Marmara zincirine bağlanabileceksiniz.

Cüzdan

- Kullanıcı arabirimi, bir cüzdan oluşturmak için iki yöntem uygulamalıdır: Kullanıcı, kendi kelimelerinden veya kullanıcı arabirimindeki getnewaddress komutunu kullanarak bir cüzdan adresi oluşturabilmelidir.
- Kullanıcı, kullanıcı arabiriminden bir pubkey oluşturabilmelidir (MBC'yi normal başlatma parametreleriyle başlat ve getnewaddress ve validateddress komutlarını çalıştır). Arka uçta üretilen hatalar, kullanıcı arayüzünde kullanıcıya sunulmalıdır.
- Kullanıcı arayüzü, kişinin kendi kelimelerinden (convertpassphrase) özel anahtar oluşturmak için 12 tohum kelimesini girmeyi etkinleştirmelidir. Kullanıcıya bu kelimelerde özel karakterler kullanmayacak şekilde uyarı gösterilmelidir. Kullanıcı arayüzü agama parolasını (agamapassphrase), adresi, pubkey'i ve wif'i döndürmelidir.
- Kullanıcı, cüzdana özel bir anahtar (importprivkey) ekleyebilmelidir. Ortaya çıkan cüzdan adresi ve pubkey (validateaddress komutundan sonra) kullanıcıya gösterilmelidir.
- Kullanıcı, cüzdanındaki bir adrese karşılık gelen özel anahtarı gösterebilmelidir.

Pubkey'i ayarlama ve MBC'yi Çalıştırma/Durdurma

- Kullanıcı, MBC düğümünü belirli bir pubkey ile çalıştırabilmelidir. Kullanıcıya ait cüzdan adresleri ve ortak anahtarlar listesinden, kullanıcı bir açık anahtar ayarlayabilmeli ve onunla MBC'yi başlatabilmelidir. MBC düğümü seçilen bir pubkey ile başlatıldığında, kullanıcıya bir bilgi görüntülenebilir.
- Kullanıcıya, zinciri yeniden indeksleme ile başlatma seçeneği sunulmalıdır.
- Kullanıcı, bir yeniden tarama seçeneğiyle MBC düğümünü başlatabilmelidir.
- Kullanıcı, MBC düğümünü durdurabilmelidir. Kullanıcıyı bilgilendiren bir bilgi görüntülenebilir.

MBC'de Düğüm ve Cüzdan ile İlgili Detaylara Ulaşma

- Kullanıcı, kendi node'unun Marmarachain durumunu, senkronizasyon durumunu, mevcut zorluk durumunu, mevcut blok yüksekliğini, en uzun zinciri, bağlantıları ve en son yenileme zamanını arayüzden görebilmeli ve yenileyebilmelidir.
- Kullanıcı, düğümü için pubkey ve wallet bakiyelerini (normal miktar, aktif miktar, toplam normal miktar ve toplam aktif miktar) görebilmelidir.
- Kullanıcı, cüzdanında bulunan toplam normal tutarı ve toplam etkinleştirilen tutarı görebilmelidir.

İş Kuralları/Kısıtlamalar/Varsayımlar (Ortak anahtarı ayarlama ve MBC'yi Çalıştırma/Durdurma)

- Kullanıcı arayüzü, karakter uzunluğu kısa olan bir pubkey ayarlanmasına izin vermemelidir. (Yazılım, kullanıcının MBC'yi başlatırken yanlış bir pubkey ayarlaması için bir araç oluşturmamalıdır.)
- Yeniden indeksleme (Reindexing) seçeneği varsayılan olarak False olarak ayarlanmalıdır.
- Yeniden tarama (Rescanning) seçeneği varsayılan olarak False olarak ayarlanmalıdır.

Koinleri Aktive ve Deaktive Etme

- Kullanıcı, kullanıcı arayüzünden paraları etkinleştirebilmeli (aktive edebilmeli, kilitleyebilmeli) ve devre dışı bırakabilmelidir (kilidini açabilmelidir, deaktive edebilmelidir). Kural: Kullanıcıdan ilgili işlemin onayı istenmelidir.

Bir Adrese Koin Gönderme

- Kullanıcı, adresi girerek doğrudan bir adrese para gönderebilmelidir.

- Kullanıcı, kişi listesinden seçim yaparak para gönderebilmelidir.
- Kullanıcının koin alma talebi yapabilmesi için kendi cüzdan adresinin QR Kodunu oluşturabilmesi gerekmektedir.

İş Kuralları/Kısıtlamalar/Varsayımlar (Bir Adrese Koin Gönderme)

- Kullanıcının Aktif Adrese coin göndermemesi için uyarı verilebilir. Kullanıcı, bu eylemin jeton yaktığına dikkat etmelidir.
- Kullanıcıya, yapılan işlem için Normal Tutarından işlem ücreti kesildiği bilgisi verilebilir.

MBC'de Staking/Madencilik Modu

- MBC düğümünün staking ve madencilik olmak üzere iki modu vardır.
- Kullanıcı arayüzü düğümünün mevcut modunu getgenerate komutu göstermelidir.
- Kullanıcı MBC düğümünün modunu değiştirebilmelidir. (setgenerate)
- Kullanıcı düğümün modunu staking olarak değiştirmek isterse, cüzdanda staking için aktifleştirilmiş bir miktar olması gerektiği konusunda kullanıcıyı uyaran bir bilgi görüntülenebilir.
- Kullanıcı düğümün modunu madencilik olarak değiştirmek isterse kullanıcı arayüzü, kullanıcıdan madencilik için kullanılacak çekirdek sayısını girmesini istemelidir. Maksimum iş parçacığı sayısını seçme seçeneği olmalıdır.

Kredi Döngüleri Operasyonları

Gelen Döngü İstekleri

- Kullanıcı belirli bir gün için gelen kredi döngüsü taleplerini inceleyebilmeli ve onaylayabilmelidir. Kural: Varsayılan son 24 saat olabilir. Varsayılan seçilmezse kullanıcı takvimden tarih/saat seçebilecektir.
- Kullanıcı, kredi döngüsü transfer isteklerini inceleyebilmelidir.
- Döngü aktarım isteklerinin yanı sıra gelen döngü isteklerinin de incelenebilmesi için txid, miktar, vade, alıcı pubkey alanlarının mevcut olması gerekir.
- Kullanıcıdan ilgili işlemi onaylaması istenmelidir. İlgili sonuç kullanıcı arayüzünde sunulmalıdır. (İtiraz durumunda işlem iptali veya onay mesajı için txid).

Kredi Döngüsü Taleplerinde Bulunmak

- Hamil (ürün/hizmeti satan kişi), datetime'da tutar, gönderen pubkey, vade bilgilerini girerek keşideciden (ürün/hizmeti ödeyen) kredi talebinde bulunabilmelidir. Hamil, gönderenin pubkey'ini (vericinin pubkey'ini) kişi listesinden seçebilecek veya pubkey'i manuel olarak ekleyebilecektir.
- Kredi döngüsünün onaylanması için hamiline bir onay mesajı gönderilmelidir. Ortaya çıkan txid, işlemi onaylaması durumunda hamiline iade edilmelidir. Aksi halde iptal edilen işlem için mesaj gelmesi gerekmektedir.

İşlemler ve Döngü Sorguları

- Kullanıcı explorer sitesinde seçtiği tarih-saat alanları arasında kendi işlemlerini (txid) görebilecektir.
- Kullanıcı, sahip olduğu aktif döngüler için MBC'nin tamamını sorgulayabilmeli ve döngü adresini, o döngü adresindeki miktarı, aktif döngülerdeki miktarı, kapalı döngülerdeki miktarı, bekleyen döngü sayısını ve kapalı döngü sayısını görüntüleyebilmelidir.
- Kullanıcı, keşideci ile hamil arasındaki kredi döngüsünün ayrıntılarını (para birimiyle

birlikte tutar, batonid, vade tarihi, saat, cirantanın pubkey'i) görüntülemek için bir txid'yi sorgulayabilmelidir.

Farklı İşletim Sistemlerinde Marmara Connector Paketleme Kılavuzu

Qt Gui Uygulamasını paketlemek için, fbs kullanılır.

Önkoşullar

PYTHON3 -> Fbs paketleme için python3.6 gereklidir
PyQt5
Son sürüm pip kütüphanesi

Linux Paketleme Yönergeleri
İşletim sisteminin güncel ve yükseltilmiş olduğundan emin olun:

```
sudo apt-get update
sudo apt-get upgrade
```

Bağımlılıklar için gereken araçları yükleyin:

```
sudo apt install git python3 python3-pip python3-venv
```

Depoyu yerel makinede istediğiniz bir konuma kopyalayın ve dalın master olarak ayarlandığından emin olun:

```
git clone https://github.com/marmarachain/marmara-connector.git
cd marmara-connector
git checkout master
```

Python3 sürümünü kontrol edin. Fbs'nin ücretsiz sürümü yalnızca Python 3.5 ve 3.6'yı destekler. Bu nedenle, fbs paketleme için python 3.6'ya geçtiğinizden emin olun.

```
python3 --version
```

Geçerli dizinde python3.6 ile sanal bir ortam oluşturun:

```
python3 -m venv venv
```

Sanal ortamı etkinleştirin:

```
source venv/bin/activate
```

Gerekli kütüphaneleri yükleyin:

```
sudo apt-get install qttools5-dev-tools
sudo apt-get install python3-pyqt5
sudo apt-get install pyqt5-dev-tools
pip install wheel
pip install pip==21.2.4
pip3 install pyqt5==5.15.4
pip install -r requirements
```

Temel PyQt uygulamasını kaynaktan çalıştırmak için aşağıdaki komutu yürütün:

```
fbs run
```

Uygulamanın kaynak kodunu bağımsız bir yürütülebilir dosyaya dönüştürmek için aşağıdaki komutu çalıştırın:

```
fbs freeze
```

Bu komut, uygulamayı orada çalıştırmak için daha sonra başka bir bilgisayara (sizinkiyle aynı işletim sistemine sahip) kopyalanabilecek target/ klasörünü oluşturur.

Linux'ta bir yükleyici oluşturmak için aşağıdaki komut kullanılır:

```
fbs installer
```

Ancak yükleyici komutu fpm'ye sahip olmanızı gerektirir. Kurulum için buradaki talimatları takip edebilir ve .deb yürütülebilir dosyasını oluşturmak için yükleyici komutunu kullanabilirsiniz. Fpm'yi yüklemek için:

```
sudo apt-get install ruby ruby-dev rubygems build-essential
gem install --no-document fpm
```

Fpm'nin doğru şekilde kurulduğundan emin olmak için aşağıdaki komutu çalıştırmayı deneyin:

```
fpm --version
```

Artık fbs installer komutunu çalıştırmak, .deb çalıştırılabilir Linux yükleyicisini oluşturacaktır.

Windows Paketleme Yönergeleri

Bağımlılıklar için gereken aşağıdaki araçları yükleyin:

```
git python3 python3-pip python3-venv
```

Komut istemini yönetici ayrıcalıklarıyla açın ve depoyu yerel makinede istediğiniz konuma kopyalayın:

```
git clone https://github.com/marmarachain/marmara-connector.git
cd marmara-connector
```

Branch’ı kontrol edin ve branch’ın master olarak ayarlandığından emin olun:

```
git branch
git checkout master
```

Python sürümünü kontrol edin:

```
python --version
```

Fbs'nin ücretsiz sürümü yalnızca Python 3.5 ve 3.6'yı destekler. Bu nedenle, fbs paketleme için pyhon 3.6'ya geçtiğinizden emin olun. Bunun için 1'de sağlanan kaynağa bakın.

.dll dosyalarını C:\Windows\System32 klasörü altına yerleştirin.

Geçerli dizinde sanal bir ortam oluşturun:

```
python36 -m venv venv
```

Sanal ortamı etkinleştirin:

```
call venv\scripts\activate.bat
```

Sanal ortam aktifken aşağıdaki adımlara devam edin:

Aşağıdaki kitaplıkları yükleyin:

```
python -m pip install --upgrade pip
python -m pip install pywin32
pip install wheel
pip install PyQt5
pip install pyqt5-tools
pip install pip==21.3.1
pip3 install pyqt5==5.15.4
pip install -r requirements
```

Temel PyQt uygulamasını kaynaktan çalıştırmak için aşağıdaki komutu yürütün:

```
fbs run
```

Uygulamanın kaynak kodunu bağımsız bir yürütülebilir dosyaya dönüştürmek için aşağıdaki komutu kullanın:

```
fbs freeze
```

Bu komut target/ dizinini oluşturur. Bu dizini başka bir bilgisayara (sizinkiyle aynı işletim sistemine sahip) kopyalayabilir ve uygulamayı orada çalıştırabilirsiniz. Windows'ta yükleyici komutunu kullanmadan önce lütfen NSIS'i yükleyin. Daha sonra aşağıdaki komutu kullanarak bir yükleyici oluşturun.

```
fbs installer
```

Bu komut installer dizinini oluşturur. Dizini installer olarak değiştirin, sağ tıklayın ve installer.nsi dosyasını NSIS ile derleyin. Bu, target dizin altında MarmaraConnectorSetup.exe dosyasını oluşturur.

Marmara Connector Ön Yüzünü Oluşturmak için QT Designer'ı Kullanma

Qt Designer'ı yükledikten sonra uygulamayı başlatın ve tasarımı geliştirmek/değiştirmek istemeniz durumunda **"guidesigner.ui"** dosyasını açın.

Açılan QT Designer ekranından istenilen geliştirme yapılabilir.

Tasarım kısmı bittikten sonra .ui dosyasının python dosyasına dönüştürülmesi gerekmektedir.

Kod kullanıcı arayüzünü python'a dönüştürün:

```
pyuic5 -x guiDesigner.ui -o guiDesign.py
```

Daha sonra **"guiDesign.py"** adında bir python dosyası oluşturulacaktır.

MainApp.py dosyasını istediğiniz herhangi bir IDE ile açın.

MainApp.py dosyasında **"guiDesign.py"** adlı dosya içe aktarılır. **"Ui_MainWindow"** sınıfa

genişletilir.

```
from guiDesign import Ui_MainWindow  # Import

class MainClassGUI(QMainWindow, Ui_MainWindow): #Extend

def __init__(self, parent=None):
    super(MainClassGUI, self).__init__(parent)
    self.setupUi(self)
```

Bu sayede GUI tasarımının her özelliğine ulaşabiliyoruz.
Aşağıdaki komutla projemizi deneyebiliriz.

```
python3 mainApp.py
```

****

Kaynaklar:

https://github.com/marmarachain/marmara-connector/wiki/
https://github.com/marmarachain/marmara-connector/wiki/Packaging-Guide-for-Marmara-Connector-for-different-OS
https://github.com/marmarachain/marmara-connector/wiki/Marmara-Connector-Software-Specifications-Document
https://github.com/marmarachain/marmara-connector/wiki/Using-QT-Designer-to-build-Marmara-Connector-Front-End
https://github.com/mherrmann/fbs-tutorial
https://www.learnpyqt.com/tutorials/packaging-pyqt5-apps-fbs/

MCL Kısa Başlangıç Rehberi

1) Marmara Connector'ü indirin ve kurun. Kurulum işlemi otomatik olarak zinciri senkronize etmenize yardımcı olacaktır. (Senkronizasyon gerçekleştiğinde connector'de "Zincir Durumu" ve "Zincir Senkron" karşısında yeşil yanacaktır, bu işlem farklı faktörlere bağlı olarak 1-2 gün kadar sürebilir. "Zincir Senkron" karşısında kaç günlük blok daha yükleneceğini gözlemleyebilirsiniz. Connector kaç gün kaldığının değil, kaç günlük blok kaldığının bilgisini verir.) Connector'de hangi işlemin yapıldığını sol alt köşedeki bildirimler aracılığı ile kontrol edebilirsiniz.

2) Senkronize olmadan staking ve mining açmak blokların bozulmasına sebebiyet verebilir. Ayrıca senkronize olmadan cüzdan adresi almak senkronizasyonu yavaşlatacaktır. Senkronizasyon sonrası "Zincir" sekmesinde "Yeni Adres Ekle" seçeneği ile adres veya adresler ekleyebilirsiniz. Sonrasında zinciri durdurun ve aldığınız cüzdan adreslerinden biri ile başlatın.

3) MCL satın alın ve Marmara Connector cüzdan adreslerinizden birine gönderin.

4) Cüzdan sekmesinden "Normal Bakiyeden Kilitle" seçeneği ile Connector adresinize gönderdiğiniz MCL'leri parçalar halinde kilitleyin. Tebrikler! Kilitlediğiniz MCL'ler ile 1x staking yapmaya başladınız.

5) Kredi Döngüleri sekmesindeki "Gelen Döngü İstekleri" ile size gelen kredi döngü isteklerini görebilir ve dilediklerinizi onaylayabilirsiniz. Bu işlemden sonra 3x gücüyle stakinge başlamış olursunuz. Aynı zamanda Kredi Döngüleri sekmesindeki "Döngü İsteğinde Bulunun" seçeneği ile döngü taleplerinizi gönderebilirsiniz, onaylanan taleplerinizdeki MCL miktarı 3x ile staking yapacaktır, tabii hamil olduğunuz sürece. Döngülerde hamili olduğunuz MCL'leri "Kredi Döngüleri-Gelen Kredi İstekleri-Kredi Döngüsü Transfer İsteği" seçeneği ile size gelen transfer istekleri arasından dilediklerinizi onaylayarak kullanabilirsiniz. Ancak o takdirde 3x staking gücü yeni hamile devredilecektir. Staking kazançlarınızı "İstatistikler-Kazanç İstatistikleri" sekmesinden görmek istediğiniz tarih aralığını seçip aratarak takip edebilirsiniz.

6) Eğer döngülere katılmayıp sadece MCL sahibi olmak veya biriktirmek istiyorsanız Komodo Wallet (Komodo Cüzdan) aracılığı ile MCL adresi sahibi olarak bunu gerçekleştirebilirsiniz. Diğer bir alternatif ise MCL ile komisyonsuz alışveriş yapabileceğiniz, ürünlerinizi satabileceğiniz, henüz test aşamasında olan "Zincir Market" uygulamasıdır.

https://marmara.io/
https://marmara.io/blog/
https://marmara.io/discord
https://www.youtube.com/c/MarmaraCreditLoops
https://github.com/marmarachain
https://t.me/MarmaraChain
https://marmara.io/docs/IMSS2019_WhitePaper_Turkce.pdf
https://coinpaprika.com/tr/converter/mcl-marmara-credit-loops/try-turkish-lira/1/
https://www.coingecko.com/en/coins/marmara-credit-loops

MCL Satın Alabileceğiniz Yerler: https://atomicdex.io/ https://safe.trade/ https://nonkyc.io/

Kripto Para Borsalarında İşlemler ve MCL Almak

MCL alım-satımı yapabileceğiniz üç adet borsa bulunmaktadır:
https://atomicdex.io/en/downloads/
Safe.trade
Nonkyc.io

Borsalarda henüz TL ile MCL arasında doğrudan takas olmadığı için alım-satımda izlenebilecek yolları anlatmaya çalışacağız. Önce TL'yi diğer aracı kripto birimlerle (BTC, USDT, LTC vb.) değiştirerek MCL alımı yapabileceğimiz borsalara transfer etmek gerekmektedir. TL ile kripto alım işlemi için Binance Türkiye, BTCTurk ve benzeri alternatifler bulunmaktadır. TL ile kripto aldığınız borsada kripto para transferi yapabilmek için kimlik onayı şartı aranmaktadır. Borsalarda kimliğinizi onaylatıp, transfer için varsa bekleme süresini de doldurduktan sonra transfer işlemini gerçekleştirebilecek duruma gelmiş olursunuz.
Üyelik işlemleri için şu ilgili sayfalara göz atabilirsiniz:

trbinance.com (Binance Türkiye) için:
https://www.trbinance.com/tr/support/faq/account-functions-guides/727a8a3f2e0f4dffa6630bc9b5e2dda2
https://www.trbinance.com/tr/support/faq/kyc/18f81821e61b4bbbb8e33ba0b6ff07d6

binance.com (Binance global) için:
https://www.binance.com/tr/support/faq/binance-uygulamas%C4%B1nda-nas%C4%B1l-kay%C4%B1t-olabilirsiniz-360042718372
https://www.binance.com/tr/support/faq/kimlik-do%C4%9Frulamas%C4%B1-nas%C4%B1l-tamamlan%C4%B1r-360027287111

btcturk.com için:
https://bilgibankasi.btcturk.com/hc/tr/articles/360011741098-BtcTurk-e-Nas%C4%B1l-%C3%9Cye-Olabilirim-
https://bilgibankasi.btcturk.com/hc/tr/articles/360011678437-BtcTurk-C%C3%BCzdan-Aktivasyonu

Biz sistemin nasıl işlediğini anlatmak açısından bu üç borsayı ekledik. Siz bu borsalarda işlemlerin nasıl yapıldığını anladıktan sonra diğer "TL - kripto para" değişimi yapan borsalarda daha uygun fırsatlar olup olmadığını kendiniz kontrol edebilirsiniz.

Kripto para borsalarında hesabınıza TL gönderdiğinizde para olduğu gibi hesabınızda görünür. (Tabii bankanız bir kesinti yapmıyorsa) (Binance TR ve BTCTurk için durum böyle, genel itibari ile de durumun aynı olduğunu söyleyebiliriz, en azından bu yazının yazıldığı tarihte böyle*) Binance globalden TL ile alım yapamazsınız, orası sizi Binance Türkiye'ye (TRBinance) yönlendiriyor bu işlem için.

TL yatırma-çekme sayfaları:
https://www.trbinance.com/blog/support-center-guides/binance-tr-hesab%C4%B1n%C4%B1za-nas%C4%B1l-t%C3%BCrk-liras%C4%B1-yat%C4%B1r%C4%B1l%C4%B1r%3F-82189e6491d9486fb04cf50588657106
https://www.trbinance.com/fees/schedule
https://kripto.btcturk.com/yardim/yatirma-cekme
https://kripto.btcturk.com/yardim/limitler-ve-kurallar

BTCTurk'te 100 TL altındaki bakiyeler ile "kripto para - TL" alım ve satım işlemi gerçekleştirilemiyor. TRBinance'ta ise "TL-kripto" işlemlerinde minumum emir miktarı 10 TL. (26.11.2023 tarihinde)

Alım-satım kuralları:
https://www.trbinance.com/trade-rule
https://bilgibankasi.pro.btcturk.com/hc/tr/articles/360012003838-Minimum-Al%C4%B1m-Satı%C4%B1m-Limitleri-Nelerdir-

TL hesabınızda göründükten sonra TL - kripto para takası ile kripto paraların kendi aralarındaki alım-satımlarda borsaların belirli bir komisyon kestiklerini aklınızda bulundurmalısınız. Bu komisyonlar güncellenebildiğinden oranları işlem yaptığınız borsanın ilgili sayfalarından takip etmekte fayda vardır:
https://www.trbinance.com/fees/schedule
https://kripto.btcturk.com/yardim/komisyonlar
https://www.binance.com/tr/fee/trading

Binance ve TrBinance hesaplarınızı eşleştirmek TRBinance'ta olmayan KMD gibi bazı birimleri almanızı sağlar. Binance transfer ile kripto paraları ücretsiz olarak Binance global hesaplarınıza aktarabilir, aynı şekilde globaldeki kripto birikimlerinizi TRBinance'taki hesaplarınıza geri çekebilirsiniz.
Binance ve Binance TR Hesabı Nasıl Eşleştirilir? :
https://www.trbinance.com/blog/5/0b697802eff64f2d800f4cd2abb7b0e6

TL ile kripto para alım işleminizi gerçekleştirdikten sonra MCL alımı için transfer işlemini gerçekleştirebilirsiniz. (Tabii borsalardaki varsa kripto para transferi için bekleme sürelerini doldurdu iseniz) Transfer işlemlerinde işlemi yapacağınız her iki borsanın desteklediği blokzincir ağları üzerinden transferleri yapabilirsiniz. Çekim (withdrawal) ücretleri düşük ağlar üzerinden işlem yapmak daha ekonomik olacaktır. Ancak düşük çekim ücretli ağlar ile işlem yapmadan önce karşı borsada bu kripto para birimiyle MCL veya benzeri satın almak isteyeceğiniz kripto paraya ulaşıp ulaşamayacağınızı iyi kontrol etmelisiniz. Örneğin safe.trade borsasına LTC ile transfer yaparsanız, LTC-BTC & LTC-USDT pazarları ile MCL-BTC & MCL-USDT pazarlarına geçiş imkanı bulabilirsiniz. Geçiş yapmayı planladığınız borsaların aktif olup olmadıklarını (çok beklemeden uygun fiyata alım-satım imkanı olup olmadığını) önceden kontrol etmekte fayda vardır.

https://www.binance.com/tr/fee/cryptoFee
https://kripto.btcturk.com/yardim/komisyonlar
https://kripto.btcturk.com/yardim/limitler-ve-kurallar

Transfer için ödeyeceğiniz çekim ücretini, yapabileceğiniz minumum çekim ücretini (withdrawal yani çekme sayfasında yazar, altında göndermeyi denerseniz bir uyarı alırsınız) ve karşı borsada o ağın olup-olmadığını kontrol ettikten sonra transfer yapacağınız borsadaki minumum yatırma (deposit) ücretini de kontrol etmelisiniz. Borsalar minumum deposit (yatırma) tutarı altındaki meblağlar için gözyaşlarına pek aldırış etmezler maalesef. MCL alım-satımında bu kontrolleri yapabileceğiniz borsaların ilgili linkleri:
https://nonkyc.io/fees
https://safe.trade/fees

Ayrıca son kontroller için bu linklerde yer almayan bilgileri ilgili koinlerin deposit ve withdrawal sayfalarından bulabilirsiniz.

Komodo Cüzdan'da ise merkeziyetsiz platformların kullanıcılara sağladığı avantajları görebilirsiniz. Alım-satım işlemlerinde piyasa yapıcılardan (makers) işlem ücreti alınmaz. Ancak, alıcılardan %0,13 oranında ücret alınmaktadır. Komodo (KMD) kullanan alıcılar için işlem ücreti %10 oranında indirimli olup, yalnızca %0,117'dir. Merkezi olmayan platformlar herhangi bir para çekme ücreti talep etmez ancak ilgili blokzincirinin doğrulayıcısına ödenen ağ maliyetlerini talep ederler.

Alım-satım işlemleri için bir karşılaştırma yapmak gerekirse Binance alıcı ve piyasa yapıcıdan %0,1 (bnb ile komisyon ödeme onayı verilerse yüz binde 75); BTCTurk'te piyasa yapıcıdan %0,08 ve alıcıdan %0,12 ; Komodo Cüzdan'da piyasa yapıcı için komisyon yok iken alıcıdan ise %0,13 (KMD bulunduruyorsa %0,117); Safe.trade alıcıdan ve piyasa yapıcıdan %0,1 (sunucu yükseltmesi-arayüz değişikliği yapmadan önce böyleydi); Nonkyc.io alıcıdan ve piyasa yapıcıdan %0,2 (NonKYC Token bulunduranlar için indirimler öngörülüyor, ayrıca nonkyc ile komisyon ödemesi açılırsa %25 indirim var) komisyon alınmaktadır.

Çekim (withdrawal) ücretlerini karşılaştırdığımızda merkeziyetsiz bir platform olan Komodo Cüzdan kullanıcılara sadece ağ maliyetlerini yansıtmasıyla açık ara önde görünüyor. Nonkyc.io tüm birimler için çekim ücretini 1 usd olarak belirlemiş durumda, nonkyc.io'nun avantajı ise likitide havuzları ile kullanıcılara sağladığı komisyon getirisi. Binance, BtcTurk ve safe.trade ise değişen oranlarla transfer ücretlerine eklemeler yaparak çekim ücreti uygulamaktadırlar. Yazıda paylaştığımız linklerden güncel ücretleri takip edebilirsiniz.

Binance'ta çekim ücreti düşük olan bazı koinlerden örnek verelim:
LTC (Litecoin ağı ile): 0,001 LTC (Yaklaşık 2 TL)
ETC (Ethereum Classic ağı ile): 0,008 ETC (Yaklaşık 4,50 TL)
RVN (Ravencoin ağı ile): 1 RVN (Yaklaşık 52 kuruş)
AVAX (Avalanche ağı ile): 0,008 AVAX (Yaklaşık 4,95 TL)
BNB (BNB Smart chain - bep20 ile): 0,0005 BNB (Yaklaşık 3,37 TL)
BTC (BNB Smart chain - bep20 ile): 0,0000074 BTC (Yaklaşık 8,07 TL)
USDT(BNB Smart chain - bep20 ile): 0,3 USDT (Yaklaşık 8,67 TL)
KMD (Smart chain ağı ile): 0,002 KMD (Yaklaşık 1,5 kuruş)

BTCTurk'te BTC ile ücretsiz çekim yapabilirsiniz. Ancak minumum çekim miktarı 0,0005 BTC'dir (Yaklaşık 543 TL) . Binance BNB Smart chain - bep20 ile BTC minumum çekim miktarı 0,000015 BTC'dir. (Yaklaşık 16,36 TL) Dolayısıyla 550 TL üzeri BTC transferlerinde BTCTurk, daha düşük miktarlar için ise Binance tercih edilebilir.

USDT gönderiminde mevcut durumda BNB Smart chain (bep20) ağı ile Binance daha avantajlı görünmektedir. (MCL alımı yapabileceğiniz borsalar için)

AVAX (Avalanche c-chain ve Avalanche x-chain ağları için) çekiminde ise 0,005 avax (yaklaşık 3,07 TL) çekim ücreti ile BTCTurk öne çıkmaktadır.

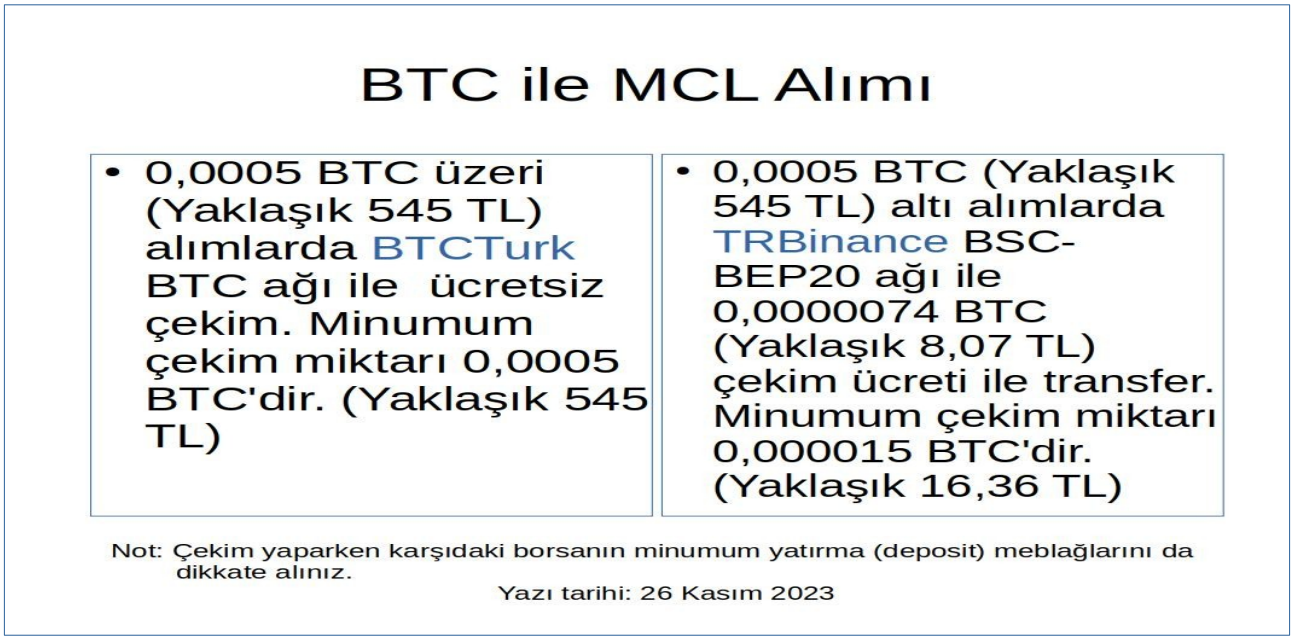
TRBinance'tan RVN alınarak diğer borsalara geçiş yapmakta ayrı bir alternatiftir. RVN'yi ilgili borsada MCL ile işlem gören birimlerden birine çevirerek MCL alımı yapılabilir.

TRBinance'tan USDT veya BTC alarak ücretsiz Binance transfer ile Binance globale geçiş yaptıktan sonra KMD alınıp transfer edilerek Komodo Cüzdan ile MCL alımı yapılabilir.

LTC düşük transfer ücretleri ile borsalar arasında ekonomik geçiş yapma olanağı sağlayan önemli koinlerden biridir.

MCL kendi zeki zincirine sahip transfer ücreti son derece düşük olan bir koindir. Komodo Cüzdan'da çekim ücreti yalnızca 0,00001 MCL'dir. (Yaklaşık 0,000005 TL) Safe.trade ise 0,01 MCL (Yaklaşık 0,005 TL) çekim ücreti uygulamaktadır. Nonkyc.io ise bildiğiniz gibi tüm koinlere standart bir ücret uygulamaktadır.

Sonuç olarak blokzincir ağları arasında sistemin nasıl çalıştığını anlarsanız kendi yolunuzu belirleyebilir, farklı alternatifleri gözden geçirebilirsiniz. Avantajlı alım-satımlar yaparak ağlar arasında kazanarak ilerleyebilirsiniz.
Biz santrayı yaptık ve pası size attık, artık top sizde:)



LTC ile MCL Alımı

* TRBinance veya BTCTurk ile LTC Alımı
* LTC Litecoin ağı ile diğer borsalara geçiş (0,001 ltc **yaklaşık 2 TL** çekim ücreti ile)
* LTC-MCL VARSA ALIM, YOKSA BTC-LTC veya USDT-LTC takası ile MCL alımına geçiş

KMD ile MCL Alımı

* TRBinance ile USDT veya BTC Alımı
* Binance transfer ile Binance globale geçiş
* KMD-USDT veya KMD-BTC takası
* KMD Komodo ağı ile Komodo Cüzdan'a transfer
* KMD-MCL takası

Not: Bu yazıdaki hesaplamalar 26 Kasım 2023 tarihindeki kurlara göre yapılmıştır.
*Bu yazı kripto paralarla ilgili yasa taslak halinde iken hazırlandı.

Sıkça Sorulan Sorulara Cevaplar

Marmara Chain kurulumunun en basit yolu, Marmara Connector'ün en son sürümünü kullanarak kurulum yapmaktır. Marmara Connector, sisteminizde Marmara Zincirinin kurulu olup olmadığını kontrol edecek, kurulu değil ise onay vermeniz halinde kurulum işlemini gerçekleştirecektir. Eğer işletim sisteminize Marmara Connector'ı kurmadan önce Marmara Chain'i kurduysanız bağlanacak ve grafiksel arayüz üzerinden MCL işlemlerinizi kolaylıkla gerçekleştirmenizi sağlayacaktır. Kurulum sürecinde herhangi bir sorunla karşılaşırsanız Discord veya Telegram hesaplarından yardım ve destek alabilirsiniz. https://github.com/marmarachain/marmara-connector/releases adresinden işletim sisteminize uygun paketi indirip yüklebilirsiniz.

Marmara kredi döngülerinin kullanılabilmesi için kullanıcının bir pubkey'e sahip olması ve bu pubkey ile zinciri başlatması beklenmektedir. Madencilik ve stake etme yoluyla kazanılan coinlerin hesaba geçebilmesi için pubkey sahibi olmak gerekmektedir. Zincir pubkey ile başlatılmazsa madencilik ve staking işlemleri boşa gidecektir.

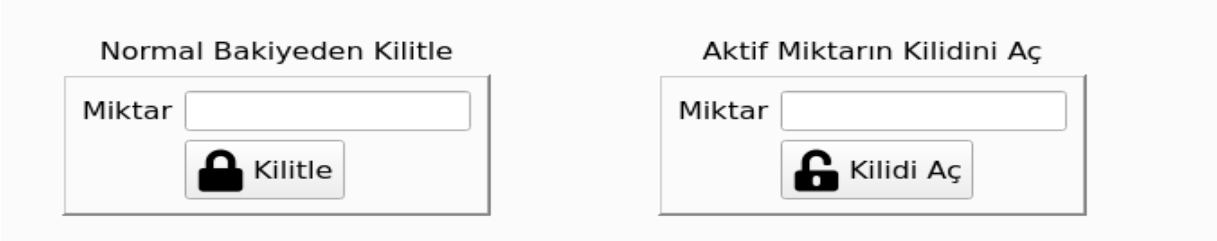
Belirlenen vade boyunca kredi döngüleri 1000'inci düğüme kadar mal ve hizmet karşılığında ekonomide dolaşabilmektedir.

MCL, kripto para borsaları arasında ekonomik olarak taşınabilir ve düşük transfer ücretleri ile başka değerlere dönüştürülebilir.

Marmara zincirinin senkronizasyon süresi, ağ bağlantı hızınıza, bilgisayarınızın işlemci gücüne, blokların kaydedildiği sürücünüzün okuma-yazma hızına ve benzeri faktörlere bağlı olarak değişir. Senkronizasyon iyi koşullarda 1 güne kadar tamamlanabilir veya biraz daha dezavantajlı koşullarda 2-3 günde gerçekleşebilir ancak minimum gereksinimleri ancak karşılayabilen bilgisayarlarda senkronizasyon süresi daha uzun sürebilir. Senkronizasyon süresini Marmara Connector üzerinden takip edebilirsiniz. Senkronizasyon süresinin uzun olabileceği durumlarda bootstrap ile süreci hızlandırabilirsiniz. (Bootstrap tavsiye edilmez, senkronizasyonun bizzat gerçekleştirilmesi daha sağlıklı olur.) Marmara Connector'de senkronizasyon sağlandığında resimde görüldüğü gibi "Zincir Durumu" (Chain Status) ve "Zincir Senkronizasyonu" (Chain Sync) karşısında yeşil yanacaktır. Senkronizasyon olmadan staking veya madencilik açmayın, veritabanını bozabilir ve tekrar senkronize etmeniz gerekebilir. Senkron olduğunda cüzdan adresi oluşturarak, hesabınıza MCL transfer etmenizi, ardından döngüler oluşturmanızı öneririz. 1x ile bulmak daha fazla zaman alır. Ödüllerin yarısı normal jeton, diğer yarısı ise aktif jetondur. Normal madeni paralar tek bloklarda, aktif madeni paralar ise çift haneli rakamlarla gösterilir. Tek haneli koinlerin içerikleri Explorer'da görünürken, çift haneli olanlar ise aktif koinlerin zeki sözleşme adresleridir. Dışarıdan Explorer'da sıfır olarak görünüyor ancak sıfır değildir. Kredi döngülerinin gücü, Marmara Chain'de stake yaparken çok daha fazladır. Bazen 3 katını bile aşabiliyor. Herkes döngüye kilitlenirse döngüler 3 katın altına düşebilir.

Normal koinlerle stake işlemi yapılamaz. Öncelikle yatırmak istediğiniz tutarı kilitlemelisiniz. Kilitleme yaparken tek seferde 300&500&1000 MCL'lik gibi küçük kısımlar halinde kilitlemeniz tavsiye edilir, benzer biçimde döngülere aktarırken hepsini bir kerede kilitlemek yerine parçalar halinde döngüler oluşturmak faydalı olacaktır. Tek seferde tek bir büyük kanca atmak yerine, daha az sayıda tek kanca atmak şansı artırır. Ayrıca döngüleri hızlandırır. Çünkü bu kilitli ödeme bilgileri

döngüler halinde kullanılmaktadır.



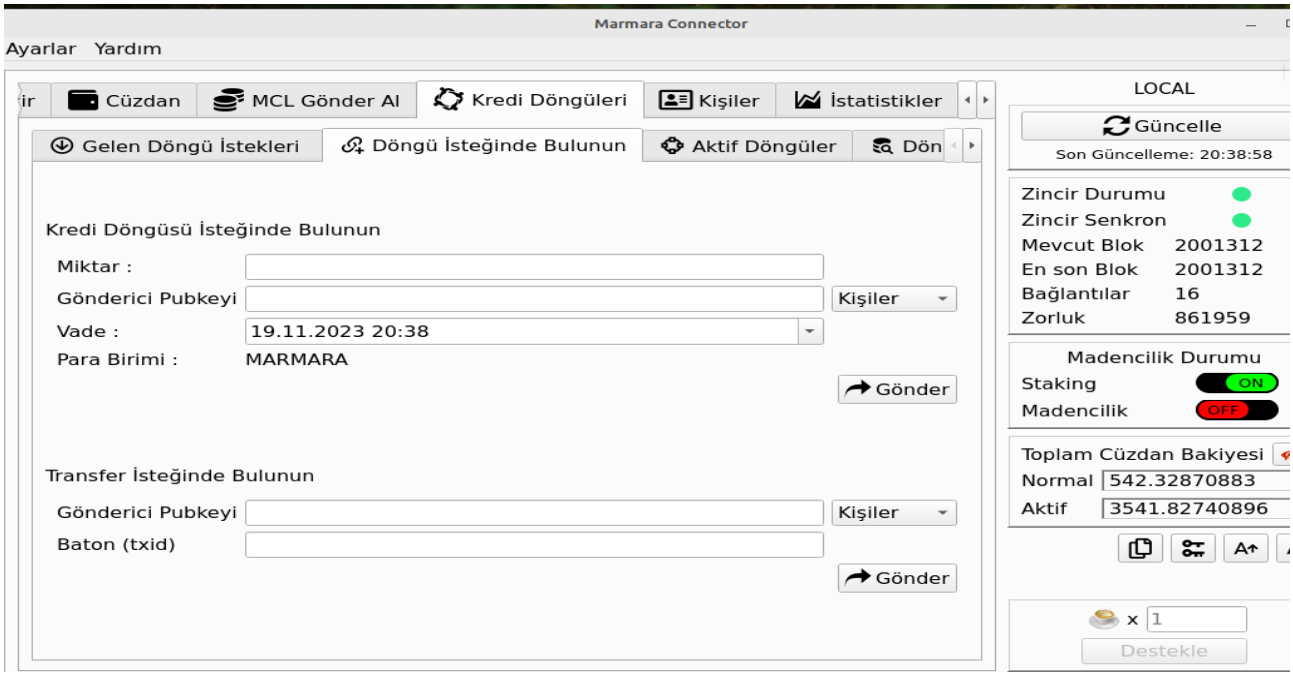
MCL'yi Marmara Connector "Cüzdan" (Wallet) sekmesi altından kolayca kilitleyebilir ve kilidini açabilirsiniz.

Stake etmeye başladıktan 8-10 gün sonra istikrarlı biçimde kazanç elde etmeye başlayabilirsiniz. Başlangıçtaki bekleme süresini bir nevi kuluçka sürecine benzetebiliriz. Ancak 300 & 500 MCL benzeri küçük parçalar halinde kilitleyip daha sonra döngülere dahil ederseniz blokları daha hızlı bulma şansınız olur. Bu şartlarda yeterli miktarda MCL ile başlarsanız ilk günden itibaren bile blok bulabilirsiniz.

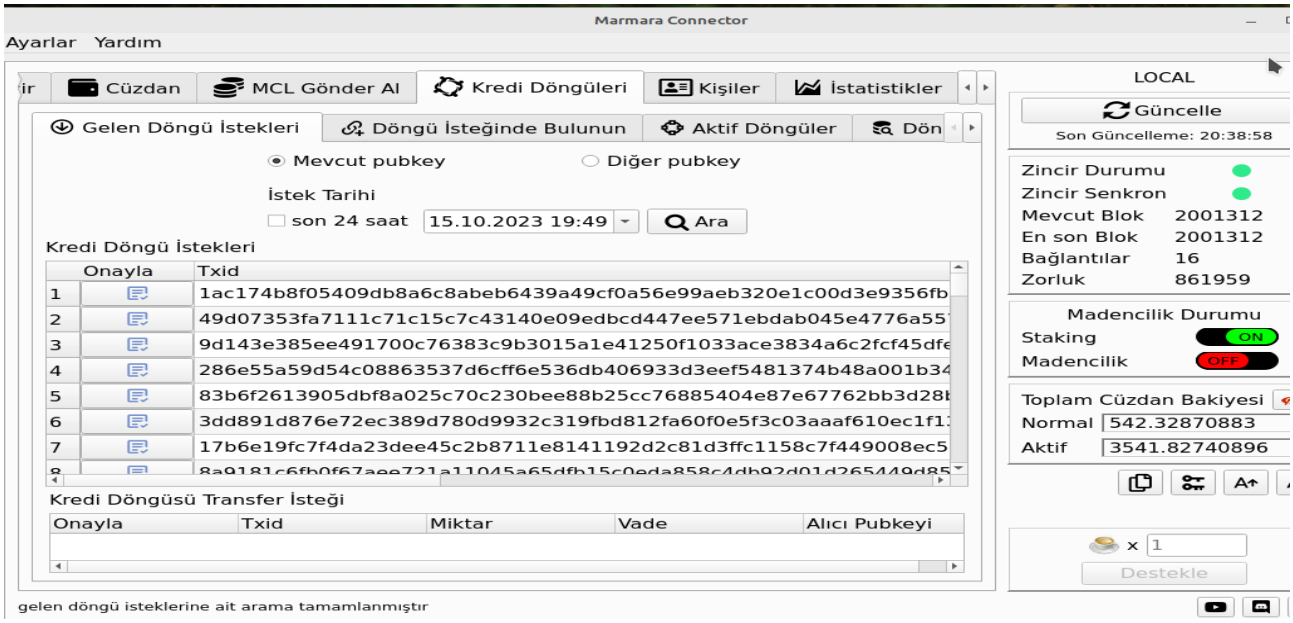
MCL ile alışveriş yapmanızı sağlayan, hatta MCL ile kendi ürünlerinizi komisyonsuz satabileceğiniz Zincir.market uygulamasına göz atmanızda fayda var.

Yüksek işlem gücüne sahip güçlü bir bilgisayarınız olmadığı sürece madencilik önerilmez. Eğer yapacaksanız bu işe uygun bir makine kullanmanız daha doğru olacaktır. Önerilen Asic madenci cihazlarıdır. Madencilik yerine daha basit bilgisayarlarla stake yaparak MCL üretebilirsiniz.

Bootstrap yedeklemesi günlük olarak ikinci tarayıcıda alınır: https://explorer2.marmara.io/bootstrap



Marmara Connector ile Marmara kredi döngüsü talebi ve transfer talebi yapabilirsiniz. “Kredi Döngüleri – Döngü İsteğinde Bulun” sekmesi ile.



Gelen Marmara kredi döngüsü talebi ve transfer taleplerini Marmara Connector “Kredi Döngüleri – Gelen Döngü İstekleri” sekmesi ile kolayca görebilir ve kabul edebilirsiniz.

Sistemdeki en son güncel durumu görebilmek için son blok durumu ile mevcut blok durumu senkronize edilmelidir. Mesela birisi size 5 MCL gönderdi diyelim, bunu hesabınızda gördünüz. Ancak ilerleyen günlerde bir sorun oluştu ve blokları tekrar sıfırdan yüklemeye başladınız. İlgili blok yüklenene kadar hesabınızdaki 5 MCL'yi göremeyeceksiniz. Yani sisteminizde gördüğünüz, yüklediğiniz son bloğun durumudur.

Ubuntu veya Linux Mint için: Eğer önceden Marmara zincirini kurarsanız, ana dizininizdeki marmara klasöründeki src klasöründe tutabilir ve böylece Marmara Connector sisteminin bulabilmesini sağlayabilirsiniz (dizin içinde komodo-cli ve komodod ile)

Staking hesaplamak için ilgili formülasyon:
([(MyActivated/TotalActivated)+(MyLoops/TotalInLoops)]*3)*8100

Marmara Connector ile kurulum işlemi sırasında özellikle Windows 10'dan eski Windows sistemlerinde curl kurulum sorunları veya zip dosyalarının açılmaması gibi sorunlarla karşılaşabilirsiniz. Bu durumlarda zip dosyasını kendiniz açarak veya curl kurulumunu manual olarak yaparak sorunu çözebilirsiniz.

Marmara Connector ile sanal sunucu üzerinde otomatik kurulum gerçekleştirebilirsiniz. Artık kodlarla uğraşmanıza gerek yok. Yerel yerine host seçeneğini seçtikten sonra IP adresi ve şifre gibi detayları girmeniz yeterli. Gerisini Connector halleder. Zincir sunucuda Connector’den bağımsız olarak çalışır. Connector sadece bağlantı yapmak, kontrol ve benzeri işleri yapmak içindir.

İleri tarihli çeklere benzer kredi döngüleri yaparak 3 kat güç ile kolayca Marmara Blockchain kurulumu ve stake yapmanızı sağlayan Marmara Connector ile kendi stake ve yatırım havuzunuzu Marmara Chain üzerinde oluşturabilirsiniz. Yeni Marmara Konnektörü, bir havuzdaki iki farklı tarih arasındaki kazançları tam olarak saymak için hassas ölçümler yapabilen bir kazanç ölçüm aracıyla birlikte geliyor. Marmara Yatırım havuzu konsepti bir uygulama ile entegre olarak projelerinize kendi kendini finanse etme özelliği eklemenizi sağlar. Örneğin yatırım havuzu ile entegrasyon sayesinde e-ticarette hem alıcı hem de satıcı komisyonlarının sıfırlanması mümkün oluyor. Marmara

Chain aynı şekilde hisse bazlı kitlesel fonlama sistemiyle veya bağımsız olarak her projeye entegre edilebiliyor. Örneğin yatırım havuzunu burs sistemiyle entegre edebilirsiniz.Yatırım havuzunda toplanan fonlar 3x staking ile entegre edilerek bursiyerlere düzenli gelir dağıtılmasında kullanılabilir. Veya bir performans sistemi ile entegre edilebilir. Tedarik Zinciri Yönetimi veya ERP uygulamaları ile entegre edilerek kendi kendini finanse etme özelliği sağlanabilmektedir.

Komut Satırından Döngüler Nasıl Başlatılır
Döngüler iki kişi arasında yapılır. Yani vermek veya almak için başka birine ihtiyacınız var. MCL ile alışveriş yapmak daha kolaydır. Alıcı aşağıdaki komut satırını yazacaktır:
marmararereceive “pubkeyofissuer” MCLamount MARMARA time '{"avalcount":"0"}'

Süre dakika cinsinden ayarlanır. Bir günde 1440 dakika olduğu için gün sayısını 1440 ile çarpmanız gerekiyor. 3 ayda 300 MCL almak istiyorsanız cirantanın pubkey'ini değiştirerek aşağıdaki komut satırını yazın:
marmararereceive “pubkeyofissuer” 300 MARMARA 129600 '{"avalcount":"0"}'

Bu size hex verecektir, sendrawtransaction HEXnumber yazmanız gerekir. Bu adım, cirantaya gönderilmesi gereken TXID'yi oluşturacaktır.

Cirantanın döngüyü onaylamak için aşağıdaki komut satırını yazması gerekir:
marmaraissue "receiver’s pubkey" '{"avalcount":"0", "autosettlement":"true", "autoinsurance":"true", "disputeexpires":"0", "EscrowOn":"false", "BlockageAmount":"0" }' TXID provided by the receiver

Ve sendrawtransaction hexnumber yazın.
Hepsi bu 🙂

Aradığınız farklı soruların cevaplarını Marmara Discord veya Marmara Telegram kanallarında bulabilirsiniz.

Linkler

Resmi İnternet Sitesi:
http://marmara.io/

Tarih:
https://marmara.io/tr/history

Takım:
https://marmara.io/tr/aboutus

Cüzdanlar:
https://marmara.io/tr/wallet

MCL Satın Alabileceğiniz Yerler:
https://atomicdex.io/
https://safe.trade/
https://nonkyc.io/

Blog:
https://marmara.io/blog/

Whitepaper:
https://marmara.io/docs/IMSS2019_WhitePaper_Turkce.pdf

Blok Gezginleri (Block Explorers):
https://explorer.marmara.io/
http://explorer2.marmara.io/
http://explorer3.marmara.io/

Coin Arzı:
https://marmara.io/tr/marmara-coin-supply

Staking Bilgi:
http://stake.marmara.io/ (Staking Hesaplayıcı)
https://marmara.io/blog/marmara-chain-1/marmara-staking-calculation-22
https://marmara.io/blog/marmara-chain-1/discover-the-superior-staking-rewards-with-marmara-mcl-coin-a-comprehensive-comparison-23

DEX Stats:
https://mcl.explorer.dexstats.info/

MCL Kurlar:
https://coinpaprika.com/tr/converter/mcl-marmara-credit-loops/try-turkish-lira/1/
https://www.coingecko.com/en/coins/marmara-credit-loops

Bootstrap https://explorer2.marmara.io/bootstrap adresinde günlük olarak yedeklenmektedir.
Bootstrap Alternatif Link: https://eu.bootstrap.dexstats.info/MCL-bootstrap.tar.gz

Diğer Linkler:
https://marmara.io/discord
https://t.me/MarmaraChain
https://marmarachain-mcl.medium.com/
https://github.com/marmarachain
https://github.com/marmarachain/marmara/releases
https://zincir.market
https://mclmarket.com/
http://www.drcetiner.org/
https://komodoplatform.com/
https://discord.com/invite/4a3J7gAmC5
https://github.com/KomodoPlatform
https://github.com/marmarachain/marmara/wiki
https://github.com/marmarachain/marmara-connector/wiki
https://www.youtube.com/@MarmaraCreditLoops/videos
https://www.youtube.com/@gultekincetiner/videos
https://www.youtube.com/@KomodoPlatformOfficial/videos
https://www.youtube.com/channel/UCBabFGXtvtThqKaYYv5hdVA/videos
https://twitter.com/marmarachain
https://www.instagram.com/marmarachainofficial/
https://www.facebook.com/groups/marmarachain
https://coinpaprika.com/coin/mcl-marmara-credit-loops/
https://www.livecoinwatch.com/exchange/atomicdex
https://cryptorating.eu/whitepapers/Komodo/2018-02-14-Komodo-White-Paper-Full.pdf
https://developers.komodoplatform.com/basic-docs/atomicdex/introduction-to-atomicdex.html
https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf
https://developers.komodoplatform.com/
http://sanaloba.org/
http://guneskoyu.org/

Kullanıcılara Önemli Uyarı

Marmara Chain
Bu yazılım, zcash'e dayanan ve deneysel olarak kabul edilen ve sürekli olarak yoğun bir şekilde geliştirilmekte olan Komodo'ya dayanmaktadır. Marmara Kredi Döngüleri deneyseldir ve devam eden bir çalışmadır. Riski size ait olmak üzere kullanabilirsiniz.

MCL ve Komodo Platform geliştiricileri hiçbir zaman bu yazılımın kullanımından kaynaklanan herhangi bir zararın sorumluluğunu kabul etmez.

Marmara Connector
Marmara Connektor deneyseldir ve devam eden bir çalışmadır. Riski size ait olmak üzere kullanın.

****

Sources:
https://github.com/marmarachain/marmara/wiki/Important-Notice-to-Users
https://github.com/marmarachain/marmara-connector/wiki/Important-Notice-to-Users

Sorun Raporlama Formatı – Marmara Connector

Lütfen sorularınızı aşağıda verilen formatta buraya göndererek bildirin:
https://github.com/marmarachain/marmara-connector/issues

Etkilenen sürümler : .....

Yeniden oluşturma adımları : ....

Mevcut davranış : .....

Beklenen davranış : ....

Video/Ekran görüntüsü bağlantısı (isteğe bağlı) :

Lisans Detayları

Marmara Chain & Marmara Connector

Bu yazılımın ve ilgili dokümantasyon dosyalarının ("Yazılım") bir kopyasını alan herhangi bir kişiye, kullanma, kopyalama, değiştirme, birleştirme hakları da dahil olmak üzere ancak bunlarla sınırlı olmamak üzere, kısıtlama olmaksızın Yazılım üzerinde işlem yapmasına işbu belge ile ücretsiz olarak izin verilmektedir. Aşağıdaki koşullara bağlı olarak Yazılımın kopyalarını yayınlayabilir, dağıtabilir, alt lisans verebilir ve/veya satabilir ve Yazılımın sağlandığı kişilerin bunu yapmasına izin verebilirsiniz:

Yukarıdaki telif hakkı bildirimi ve bu izin bildirimi, Yazılımın tüm kopyalarına veya önemli bölümlerine dahil edilecektir.

YAZILIM, TİCARİ ELVERİŞLİLİK, BELİRLİ BİR AMACA UYGUNLUK VE İHLAL ETMEME GARANTİLERİ DAHİL ANCAK BUNLARLA SINIRLI OLMAMAK ÜZERE AÇIK VEYA ZIMNİ HERHANGİ BİR GARANTİ OLMAKSIZIN "OLDUĞU GİBİ" SAĞLANIR. HİÇBİR DURUMDA YAZARLAR VEYA TELİF HAKKI SAHİPLERİ, YAZILIMDAN VEYA YAZILIMLA BAĞLANTILI OLARAK VEYA YAZILIMDAKİ KULLANIM VEYA DİĞER İŞLEMLERDEN KAYNAKLANAN, SÖZLEŞME, HAKSIZ FİİL VEYA BAŞKA BİR EYLEMDE OLSUN, HERHANGİ BİR İDDİA, ZARAR VEYA DİĞER YÜKÜMLÜLÜKLERDEN SORUMLU TUTULAMAZ.

****

Kaynaklar:
https://github.com/marmarachain/marmara/wiki/License-Details
https://github.com/marmarachain/marmara-connector